Homecomputer

4 1. Jahrgang

April '83 **5,50 DM** 45 öS 6,00 sfr

In diesem Heft u.a.:

ZX-81
Life
Survive

VC-20
Spukhaus
Zeichengenerator II
Assembler

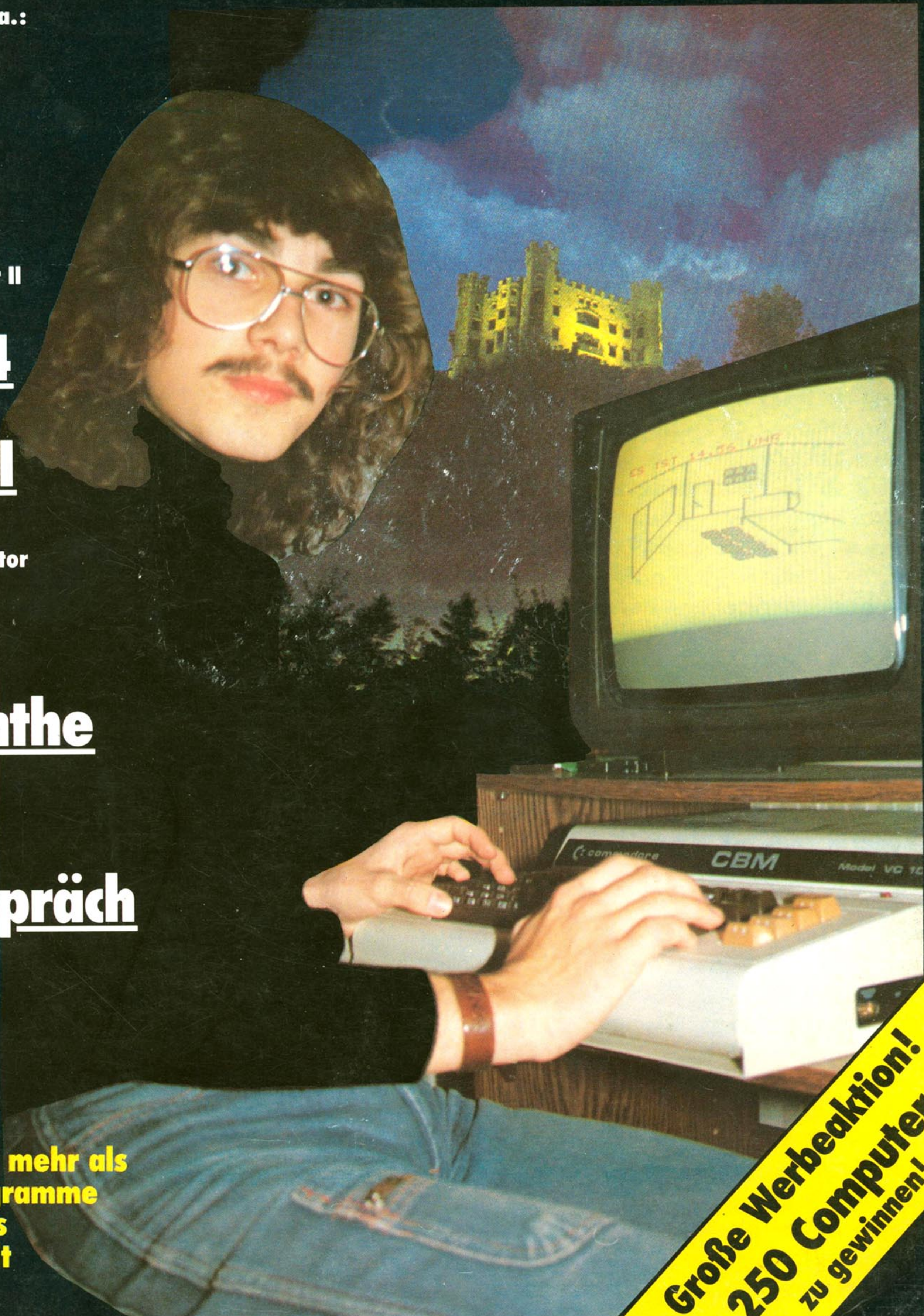
TI-99/4
Asteroiden

Apple II
Diskettenschutz
Oktopus
Suchrätselgenerator

VC-64
Kalaha

Labyrinthe
Apple II
ZX-81
VC-20
CBM 3008

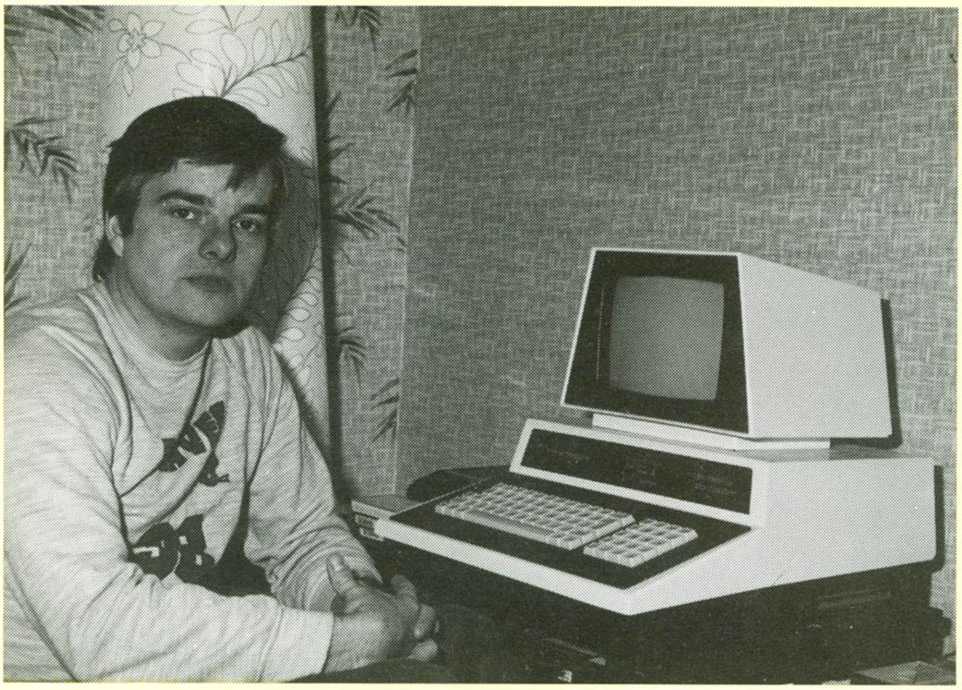
Im Gespräch
Gregor Widuch



Jeden Monat mehr als 15 neue Programme + Tips + News aus aller Welt

Große Werbeaktion! 250 Computer zu gewinnen!

Homecomputer wird ausschließlich unter Verwendung von Microcomputern erstellt



Liebe Leserin, lieber Leser

Herzlichen Dank, für die vielen telefonischen und schriftlichen Glückwünsche zum Erscheinen unserer Zeitschrift Homecomputer.
Schneller als erwartet kamen Aufgaben auf uns zu, die wir mit unseren kleinen Mitarbeiterstab kaum erledigen konnten.
Deshalb sind auch einige Dinge in diesem Heft anders, als sie vor einem Monat geplant waren.
Dank der Unterstützung der Druckerei, die unser Magazin druckt, sind wir in der Lage, äußerst schnell umzudisponieren und Artikel ins Heft aufzunehmen, die erst wenige Tage bevor das Heft im Handel vorrätig ist, geschrieben wurden.
Darum können wir auch schon die ersten Leserbriefe abdrucken, die für uns besonders interessant waren.
Nicht immer war die Kritik sachlich, aber im Großen und Ganzen können wir mit der Reaktion auf unser erstes Heft zufrieden sein. Etwa 95 % aller Briefschreiber äußerten sich positiv, was unser Team natürlich besonders erfreute.
Ich glaube, wir haben die richtige Richtung eingeschlagen, und hoffe, daß wir Ihnen auch weiterhin das bieten können, was Sie von einer Zeitschrift für Homecomputer-Anwender erwarten.
Daß im ersten Heft einige Programme nicht auf Anhieb liefen, bitte ich zu entschuldigen. Wir bemühen uns, daß so etwas in den folgenden Ausgaben nicht mehr vorkommen kann. Gelegentlich traf uns aber keine Schuld und wir mußten uns böse Beschimpfungen anhören. Nach kurzem Gespräch sahen die Anrufer dann ein, daß sie etwas zu heftig reagiert hatten.
Und noch eine Sache fällt mir gerade ein. Bei der Abonnentenwerbeaktion haben einige Leser übersehen, wer teilnahmeberechigt ist, nämlich nur derjenige, der schon Abonnent von Homecomputer ist. Es geht also nicht, daß die Freundin oder die Frau ihren Angebeteten wirbt und dadurch am Wettbewerb teilnimmt. Auf diese Weise wäre es nicht möglich, solch eine Aktion zu starten.
Wir hoffen, daß uns die so geworbenen Abonnenten nicht böse sind, und trotzdem unsere Leser bleiben.
Übrigens, der Wettbewerb geht in die erste entscheidende Phase. Wenn der Trend sich bestätigt, werden bald die ersten Computer verteilt werden können.

Herzlichst
Ihr Ralph Roeske



Herausgeber, Chefredakteur

Inhalt

Impressum

Homecomputer erscheint monatlich im:
Roeske Verlag, Eschwege

HERAUSGEBER:
Ralph Roeske

REDAKTION:
Ralph Roeske (R), Chefredakteur (verantwortlich)
Horst Brand (Br)
Cornelius Geppert (Ge)

HERSTELLUNG:
Roeske-Verlag
Eschwege

REPRODUKTION:
Intal Composing GmbH
3436 Hessisch Lichtenau

DRUCK:
Vogt GmbH
Druckerei + Verlag
3436 Hessisch Lichtenau

VERTRIEB:
Inland (Groß-, Einzel- und Bahnhofsbuchhandel), sowie Österreich und Schweiz:
Verlagsunion
Friedrich-Bergius-Str. 7
6200 Wiesbaden
Tel.: 06121-2660

ANZEIGENLEITUNG:
Monika Roeske (M)

ERSCHEINUNGSWEISE:
Erstverkaufstag von Homecomputer ist jeweils der letzte Montag vor dem Erscheinungsmonat

ANSCHRIFT:
Homecomputer
Gartenstraße 47
3440 Eschwege
Tel.: 05651-5993

BEZUGSPREISE:
Einzelheft: 5,50 DM
Abonnement: Inland 55,- DM pro Jahr (12 Ausgaben)
Ausland: Europa 80,-DM
USA 110,-DM

ANZEIGENPREISE:
Es gilt die Anzeigenpreisliste Nr.1 vom 1.Januar 1983
1/1 Seite S/W 3000,-DM
Farbzuschlag je Farbe (Europaskala) 500,-DM

AUTOREN, MANUSKRIPTE:
Der Verlag nimmt Manuskripte und Software zur Veröffentlichung gerne entgegen. Honorare nach Vereinbarung. Bei Zusendung von Manuskripten und Software erteilt der Autor dem Verlag die Genehmigung zum Abdruck.
Rücksendung erfolgt nur bei angeforderten Beiträgen, ansonsten nur gegen Erstattung der Unkosten.
Zusendungen von Software zur Veröffentlichung sollte bitte folgendes enthalten: Kopierfähige Kassette oder Diskette mit dem Programm, von Drucker erstelltes Listing oder Serie von Bildschirmfotos (keine Schreibmaschinen-Listings), eventuell Bildschirmfotos von einem Probelauf.

Vom Spiel zum Beruf?



Etwa 2 Wochen nach Erscheinen der ersten Ausgabe von Homecomputer erreichte die Redaktion die Zuschrift eines 14-jährigen Schülers, der mit Hilfe des darin abgedruckten Zeichengenerators innerhalb kürzester Zeit ein spannendes Adventure-Programm geschrieben hat, das wir in diesem Magazin vorstellen.

Wir gingen daraufhin der Frage nach, wie ein Junge das Computer-Hobby entdeckt und sich in diese Materie eingearbeitet hat.

Homecomputer (HC): Gregor, du wohnst in einem kleinen Ort in Nordhessen und besuchst die 8. Klasse einer Realschule. Wie bist du zu deinem VC-20 gekommen, wie fing das ganze an?

Gregor Widuch (G): Ein Freund meiner Eltern, den wir häufig am Wochenende besuchten, war begeisterter Computer-Fan und besaß damals einen Commodore Pet. Das war so in der Zeit von 1979-1980. Jedesmal wenn wir dort waren, hatte er neue Programme, auf die wir uns mit Begeisterung stürzten. Nach einiger Zeit fing ich an, mich auch für das Programmieren selbst zu interessieren und begann, ganz einfache Programmschritte zu verstehen zu lernen. Im Laufe der Zeit ist dann hieraus der Wunsch entstanden, selbst einen Computer zu besitzen.

HC: Was sagten denn deine Eltern zu diesem Wunsch? Vielleicht will dein Vater, der hier neben uns sitzt, selbst etwas hierzu sagen?

Vater (V): Natürlich waren meine Frau und ich am Anfang skeptisch. Wir sahen zwar das Interesse, das der Junge der Sache entgegenbrachte, waren uns aber nicht sicher, ob dies von längerer Dauer sein würde und ob er nicht zu jung hierfür wäre.

Schließlich waren Heimcomputer zu dieser Zeit, nämlich 1981, noch recht teuer und ich bin kein Krösus. Da dieser Wunsch all-



mählich jedoch bei unserem Sohn zu einer fixen Idee wurde, überraschten wir ihn Weihnachten 1981 doch mit einem VC-20 in der Grundausstattung. Ein kleiner Schwarz/Weiß-Fernseher als Monitor war schon vorhanden. Bei dieser Entscheidung haben wir das pro und contra sorgfältig abgewogen und sind zu der Überzeugung gelangt, den Versuch zu wagen. Viel schaden konnte es ja nicht. Schwierig war für uns als Laien die Wahl des geeigneten Gerätes. Wir entschieden uns nach Beratung mit meinem Freund für den VC-20, der bedienungsfreundlich, leicht verständlich und mit Blick auf die Zukunft, auch erweiterungsfähig erschien.

HC: Gregor, nachdem du endlich deinen VC-20 bekommen hattest, bist du dann sofort damit zurecht gekommen oder hattest du auch die üblichen Anfangsschwierigkeiten?

G: Als ich am Weihnachtsabend den Computer unter dem Baum entdeckte, war ich natürlich erst einmal total happy. Ich hatte nicht gedacht, daß ich dieses Geschenk tatsächlich bekommen würde. Die ersten Versuche waren allerdings totale Versager. Über den SYNTAX ERROR und andere simple Fehler kamen mein Vater und ich in den ersten Tagen kaum hinaus. Wir haben das dünne Anleitungsheft zig-mal durchgelesen, trotzdem kam uns alles spanisch vor. Dann haben wir angefangen, die Lernbeispiele von Commodore einzutasten und waren schließlich mächtig stolz, das Miniprogramm "Vogel flattert über Bildschirm" funktionieren zu sehen.

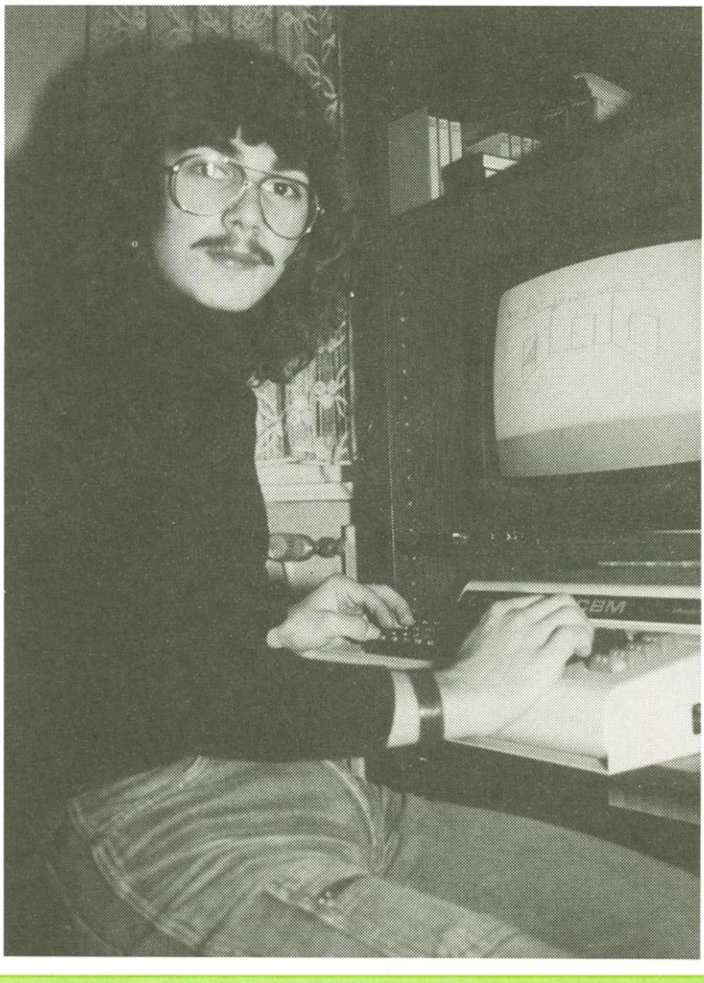
V: Ja, das stimmt! Nach einigen Wochen war ich dann allerdings doch erstaunt, daß Gregor schon einfache kleine Programme selbst zustande brachte. Er war stolz über jeden Fortschritt, den man fast täglich beobachten konnte.

HC: Wie hast du nun tatsächlich programmieren gelernt?

G: Nachdem ich erst einmal die einzelnen Funktionen des Apparates heraus hatte, fing ich mit den PRINT-Befehlen an, die ich immer weiter ausbaute. Danach kam die FOR-NEXT-Schleife und irgendwann später POKE. Aber das hat dann schon etwas länger gedauert. Zwischendurch habe ich natürlich auch Programme aus Magazinen eingetippt und versucht diese zu analysieren und zu verbessern.

V: Anfangs war Gregors Interesse mehr auf das Spielen ausgerichtet. Im Laufe der Zeit, ich würde sagen so etwa nach einem halben Jahr, verschoben sich seine Interessen allerdings immer mehr in Richtung Programmieren. Jetzt ist es fast schon so, daß ein Spiel, ist es erst einmal richtig fertig, ihn nicht mehr sonderlich interessiert. Oft bin ich es, der ihm dann die Idee zu einem neuen Programm liefert.





HC: Und wie ging es dann weiter?

G: Nach einiger Zeit wurden meine Programme immer größer und die 3,5 K-Byte des VC-20 waren einfach zu wenig. Mein Vater hat das ja eingesehen; schwieriger war es, meine Mutter von der Notwendigkeit einer Speichererweiterung zu überzeugen. Ich habe dann von meinem Sparbuch Geld abheben dürfen und so die 16-K-Erweiterung bekommen.

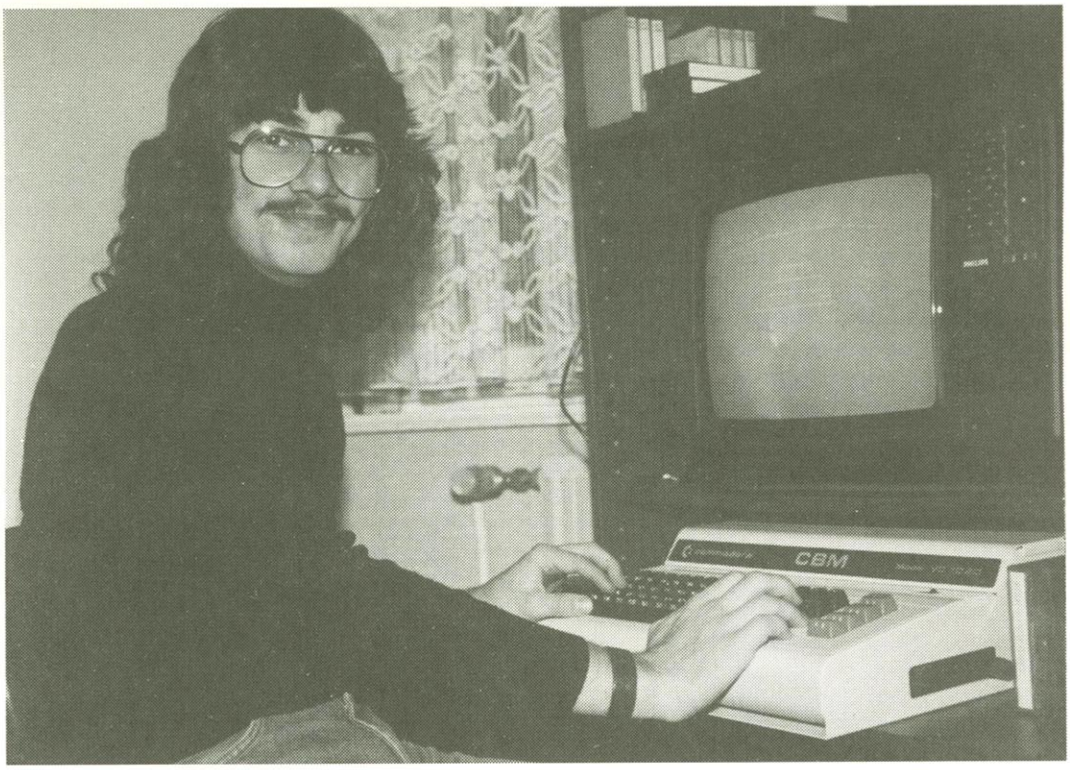
HC: Nun eine andere, wichtige Frage. Welche Auswirkungen hat dein neues Hobby denn auf deine schulischen Leistungen gehabt und hast du neben dem Computer noch andere Hobbys?

Im Gespräch

G: Na ja, im letzten Jahr hat sich mein Notendurchschnitt sogar verbessert, das habe ich mit dem Computer errechnet. Ich glaube für Mathe und Englisch (wegen der vielen englischen Ausdrücke in den Programmen) war der VC-20 schon ganz nützlich. Natürlich habe ich auch noch andere Hobbys, der Computer ist allerdings Nummer 1! Ich höre gerne Musik (Hardrock und solche Sachen), sammle Briefmarken, Münzen und allen möglichen anderen Kram. Für Sport interessiere ich mich weniger.

HC: Beurteilen sie als Vater das auch so wie ihr Sohn oder sehen sie die schulische Seite eventuell anders?

V: Ich glaube, im Großen und Ganzen kann ich den Aussagen von Gregor zustimmen. Ab und zu muß man seinen Eifer allerdings etwas bremsen und ihm klarmachen, daß Schularbeiten noch immer Vorrang haben. Ansonsten hat der Computer bestimmt keinen schlechten Einfluß auf ihm. Mittlerweile ist es schon zu einem festen Berufswunsch bei ihm geworden, etwas in der Computerbranche zu erlernen.



HC: Gregor, was sagen deine Freunde zu deinem Hobby oder wissen die nichts von dem, was du in deinem stillen Kämmerlein betreibst?

G: Die meisten aus meiner Klasse kennen meinen Computer und spielen gerne damit. Ein Freund von mir möchte demnächst auch gerne einen Computer haben.

V: Ja, ich kenne die Eltern dieses Jungen und wir haben über die Angelegenheit gesprochen. Die Schwierigkeit besteht darin, daß jemand, der keinen Einblick in diese Materie hat, sich kaum vorstellen kann, daß ein 12-14 jähriger durchaus ernsthaft mit einem Computer umgehen lernen kann. Es ist für die meisten Kinder sicherlich einfacher ihre Eltern von der Notwendigkeit einer teuren Hi-Fi-Anlage oder eines Mopeds zu überzeugen, als von einem Computer. Und das, obwohl diese Dinge teurer als ein Microcomputer in der Grundausstattung sind.



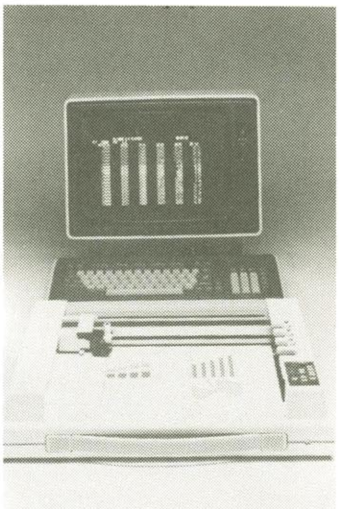
HC: Da haben sie sicher recht. Wir haben schon öfter betont, daß dieses interessante Hobby bei uns noch in den Kinderschuhen steckt. Gregor, was sind deine weiteren Pläne in dieser Richtung?

G.: Nachdem ich letzes Jahr zu Weihnachten einen Farbmonitor bekommen habe, müßte als nächstes eine Floppy-Disk her, da das ständige Laden mit dem Cassettenspeicher auf die Dauer recht zeitraubend ist.

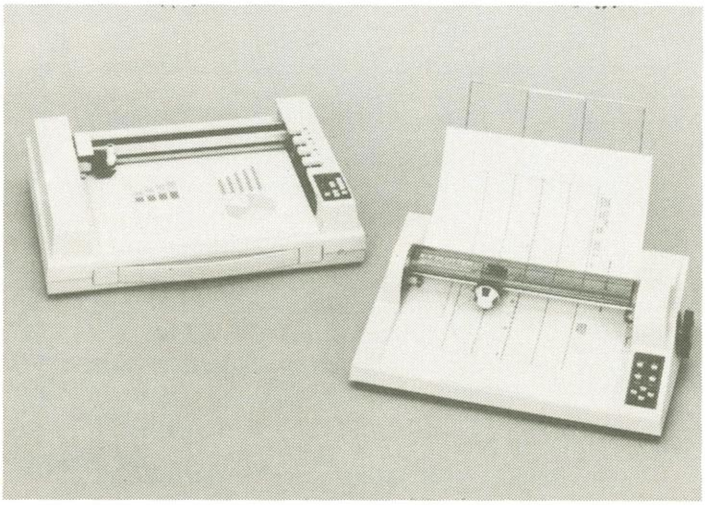
HC: Dann wird ja das Honorar für dein Spukhaus-Programm gut angelegt werden, oder? (nickt) Mach weiter so!

Digitalplotter-Serie

YEW, Japans führender Hersteller von Registriergeräten, stellt nun auch eine neue Serie von Plottern vor. Diese besteht aus den zwei Gerätefamilien PL-1000 und PL-2000.
Der PL-1000 ist ein äussersts kompakter 4-Farben DIN-A3-Plotter. Dank seiner Funktionsweise, ähnlich der eines Trommelplotters, hat er nur die halben Abmessungen eines Flachbettplotters. Die Standardausführung hat bereits eine Vielzahl von intelligenten Funktionen (23 Standard-Grafik-Kommandos), wie z.B. Kreise, Keissegmente, Achsen, Charakter und Symbole in verschiedenen Größen und beliebiger Richtung, 9 verschiedenen Linientypen, usw. Er ist mit 2 Schnittstellen lieferbar, 8 bit parallel (Centronics) und RS-232 C.
Der PL-2000 ist ein hochauflösender (50 um) 4-Farben DIN-A4-Flachbettplotter in hervorragender Qualität. Er ist in der Grundausführung noch intelligenter als der PL-1000 (34 Standard-Grafik-Kommandos). Er beinhaltet z.B. die Digitalisierungsfunktion, die Ausgabe der Federposition zum Rechner. Dieser Plotter ist mit 3 Schnittstellen lieferbar, 8 bit parallel (Centronics), RS-232 C und GP-IB (IEEE-625).
Beide Geräte sind auch zum Schreiben auf Folie ausgerüstet.



Als Option gibt es ein Graphik-ROM, das die Einsatzmöglichkeiten noch wesentlich erweitert. Es enthält zusätzliche 22 Grafikbefehle, wie z.B. Balkendiagramme, Tortendiagramme, Liniendiagramme, usw.
Info: abn ELEKTRONIK, Herrsching



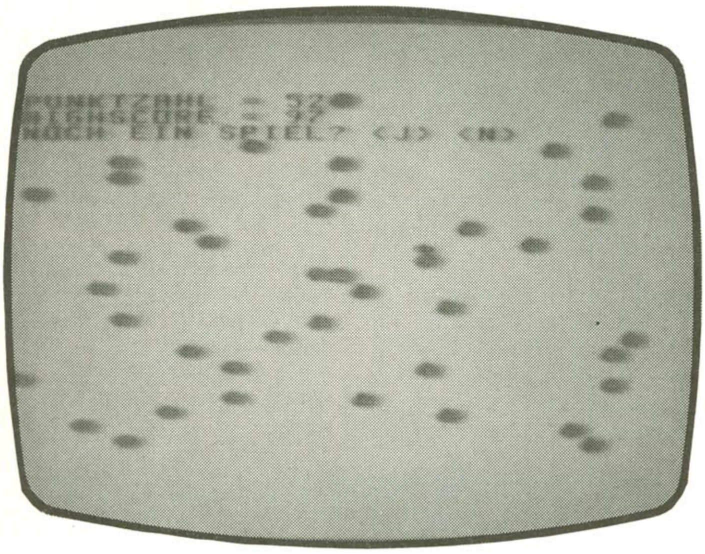
Soft- und Hardware aus England importiert

Anwender der populären englischen Homecomputer wie Sinclair ZX-81 und Spectrum, Oric, Dragon, BBC-Micro usw. können sich freuen. Immer mehr Firmen gehen dazu über, Software und Peripherie aus England zu importieren und auf dem deutschen Markt anzubieten.
Die gesamte Pallette von Quicksilva und Pixel ist ab sofort in unseren Lande direkt zu erwerben. Space Intruders oder Meteor Storms, Harvester, Starquest, Trader oder Zor - um nur einige zu nennen - werden nicht nur importiert werden, sondern auch in Kürze mit deutscher Beschreibung und Anleitung angeboten werden.
Auch die lange erwarteten Spectrum Joysticks können dann ohne Umweg bezogen werden.
Info: Freiumschlag an Bill Read, Postfach 66, 3388 Bad Harzburg

Dallas geht, Texas kommt

Sein Basic ist zwar äusserst langsam, aber trotzdem hat der TI-99/4A viele Anwender gefunden.
Bevor wir im Mai-Heft zum großen Texas-Festival mit 5 TI-99-Programmen einladen, in diesem Heft wieder ein einfacheres Reaktionsspiel:

Asteroiden



Zwei Tasten (Q und A) stehen dem Spieler zur Verfügung, um sein UFO durch den Asteroidengürtel zu steuern.
Das Programm läuft auf allen TI-99/4A - das Extended Basic Modul ist nicht erforderlich.
Der Spielaufbau ist sehr einfach.
Bis zur Zeile 160 werden die Zeichen definiert und eingefärbt.
Ab Zeile 170 beginnt dann das eigentliche Spiel. Es werden per Zufallszahlen Asteroiden auf den Bildschirm gebracht, die das Raumschiff des Spielers umfliegen muß.
Wird es getroffen, so endet das Spiel, was ebenfalls passiert, wenn zu hoch oder zu niedrig geflogen wird.
Die Zeilen 800 bis 1190 sind mit denen im Spiel aus Heft 3/83 identisch (bis auf die Korrektur zum richtigen Bildschirmausdruck). Hier wird wieder mittels der HCHAR-Funktion Text auf den Bildschirm geschrieben.

```
100 RANDOMIZE
110 CALL CLEAR
120 CALL CHAR(96,"0000000018
7E997E")
130 CALL CHAR(152,"3C7EFFFFF
F7E3C")
140 CALL CHAR(153,"1C3E7F7FF
FFF7E3C")
150 CALL COLOR(9,2,1)
160 CALL COLOR(16,9,1)
170 PZ=0
180 NUFOX=1
190 NUFOY=12
200 UFOX=NUFOX
210 UFOY=NUFOY
220 A1=RND*14+1
230 A2=RND*15+15
240 CALL GCHAR(UFOY+1,UFOX,T
)
250 IF T=152 THEN 800
260 IF T=153 THEN 800
270 PZ=PZ+1
280 CALL HCHAR(UFOY,UFOX,32,
1)
290 UFOX=NUFOX
300 UFOY=NUFOY
310 CALL HCHAR(24,A1,152,1)
320 CALL HCHAR(24,A2,153,1)
330 PRINT
340 CALL HCHAR(NUFOY,NUFOX,9
6,1)
350 CALL KEY(0,K,ST)
360 IF ST=0 THEN 390
370 IF K=81 THEN 430
380 IF K=65 THEN 460
390 NUFOX=NUFOX+1
400 IF NUFOX<33 THEN 420
410 NUFOX=1
420 GOTO 220
430 NUFOY=NUFOY-1
440 IF NUFOY<1 THEN 800
450 GOTO 390
460 NUFOY=NUFOY+1
470 IF NUFOY>23 THEN 800
480 GOTO 390
800 DRUCK$="PUNKTZAHL = "&ST
R$(PZ)
810 VTAB=1
820 HTAB=2
830 GOSUB 960
840 IF HS>PZ THEN 870
850 HS=PZ
860 GOSUB 1140
870 DRUCK$="HIGHSCORE = "&ST
R$(HS)
880 GOSUB 960
890 DRUCK$="NOCH EIN SPIEL?
<J> <N> "
900 GOSUB 960
910 CALL KEY(0,KEY,STATUS)
920 IF STATUS=0 THEN 910
930 IF KEY=74 THEN 190
940 IF KEY<>78 THEN 910
950 END
960 REM PRINTROUTINE
970 FOR DRI=1 TO LEN(DRUCK$)
980 CODE=ASC(SEG$(DRUCK$,DRI
,1))
990 GOSUB 1030
1000 NEXT DRI
1010 GOSUB 1080
1020 RETURN
1030 REM COUT
1040 CALL HCHAR(VTAB,HTAB,CO
DE)
1050 HTAB=HTAB+1
1060 IF HTAB<32 THEN 1130
1070 REM CROUT
1080 VTAB=VTAB+1
1090 HTAB=2
1100 IF VTAB<25 THEN 1130
1110 VTAB=24
1120 PRINT
1130 RETURN
1140 FOR SI=1 TO 5
1150 FOR SJ=-3 TO -1
1160 CALL SOUND(100,SJ,1)
1170 NEXT SJ
1180 NEXT SI
1190 RETURN
```

Kalaha

das Bohnenspiel

Eines der reizvollsten Spiele, die es überhaupt gibt, ist das jahrtausende alte afrikanische Bohnenspiel.

Um Haus und Hof ging es, wenn auf Stammestreffen die Häuptlinge der einzelnen Dörfer gegeneinander im geistigen Wettkampf antraten.

Bei Kalaha geht es darum, in seine Heim-Mulde die meisten Bohnen abzulegen. Dieses kann man immer dann tun, wenn ein Verteilungsvorgang eine Menge von 2, 4 oder 6 Bohnen in der Mulde ergibt, in der die letzte Bohne abgelegt wird. Ist auch die jeweils vorletzte Mulde mit 2, 4 oder 6 Bohnen gefüllt, so wird diese ebenfalls geleert und ihr Inhalt in die Heim-Mulde gelegt.

Der Verteilungsvorgang sieht so aus: Der Spieler, der am Zug ist, entscheidet sich für eine seiner 6 Mulden, nimmt alle Bohnen heraus und verteilt sie einzeln auf die, im Gegenuhrzeigersinn folgenden, wobei er auch vor den Mulden seines Gegners nicht Halt macht. Es herrscht Zugzwang.

Sind die Mulden einer Seite leer, so endet das Spiel. Das Gleiche geschieht, wenn eine Seite mehr als 36 Bohnen in ihre Heim-Mulde bringen konnte.

Das Spiel gegen den Computer ist nicht einfach, obwohl er nur, je nach Spielstärke, einen oder zwei Halbzüge 'voraussehen' kann.

```
1 POKE53280,9:POKE53281,9
5 PRINT"■ ■■■PGM: KALAHA":PRINT:PRINT
6 PRINT"■■■■■■■■■■■"
10 DIMSB(12),A1(12),A2(12),ER(6)
50 FORI=1TO12:SB(I)=6:NEXT
51 INPUT"WOLLEN SIE ANFANGEN (J/N)";JN$
52 PRINT:INPUT"SCHWIERIGKEITSGRAD (1-2)";SG
53 IFSG<1ORSG>2THEN52
55 PRINT"■"
60 GOSUB2000
70 IFJN$="J"THEN1804
90 E1=-1000
95 FORI=1TO6:ER(I)=-1000:NEXT
100 FORI=1TO6:ER(I)=-1000:NEXT
110 FORI1=1TO6:G1=0:IFSB(I1)=0THEN1600
115 FORI=1TO12:A1(I)=SB(I):NEXT
120 AN=A1(I1):A1(I1)=0
130 FORJ=1TOAN:IN=I1+J-INT((I1+J)/12.01)*12:A1(IN)=A1(IN)+1:NEXT
200 AI=A1(IN):IFAI=2ORAI=4ORAI=6THEN210
205 GOTO300
210 G1=G1+AI:A1(IN)=0:IN=IN-1:IFIN=0THENIN=12
220 GOTO200
300 REM
310 IFG1+HC>36THENZU=I1:GOTO1620
330 E2=1000
335 IFSG=1THENER(I1)=G1:E1=0:GOTO1600
340 FORI2=7TO12:IFA1(I2)=0THEN500
342 G2=G1
345 FORJ=1TO12:A2(J)=A1(J):NEXT
350 AN=A2(I2):A2(I2)=0
360 FORJ=1TOAN:IN=I2+J-INT((I2+J)/12.01)*12:A2(IN)=A2(IN)+1:NEXT
370 AI=A2(IN):IFAI=2ORAI=4ORAI=6THEN380
375 GOTO400
380 G2=G2-AI:A2(IN)=0:IN=IN-1:IFIN=0THENIN=12
390 GOTO370
400 IFG2<E2THENE2=G2
410 IFG1-G2+HS>=36THENE2=-750
500 NEXTI2
1000 ER(I1)=E2:E1=E2
1100 IFE2=1000ANDHS>HCTHENER(I1)=-500
1600 NEXTI1
1610 IFE1=-1000THEN3000
1611 FORI=1TO6:ZU(I)=I:NEXT
1612 F=0:FORI=2TO6:IFER(I)=<ER(I-1)THEN1614
1613 ER=ER(I):ER(I)=ER(I-1):ER(I-1)=ER:IZ=ZU(I):ZU(I)=ZU(I-1):ZU(I-1)=IZ:F=1
1614 NEXT
1615 IFF=1THEN1612
1616 F=1:FORI=2TO6:IFER(I)=ER(1)THENF=F+1
1617 NEXT
1618 ZU=ZU(1):IFF>1THENZU=ZU(INT(RND(1)*F+1))
1620 PRINT"■                                        ":PRINT"■MEIN ZUG LAUTET";ZU;"(";SB
(ZU);")"
1700 GOSUB2100
1760 AI=SB(IN):IFAI=2ORAI=4ORAI=6THEN1770
1765 GOTO1800
1770 HC=HC+AI:SB(IN)=0:IN=IN-1:IFIN=0THENIN=12
1780 GOSUB2000
1790 GOTO1760
1800 U=0:FORI=7TO12:IFSB(I)=0THENU=U+1
1802 NEXT
1803 IFU=6THEN3000
1804 PRINT"■■■■■■■■■■■■■■■■          ";:IFHC>36ORHC+HS=72THEN3000
1805 INPUT"■■■■■■■■■■■■IHR ZUG";ZU
1806 IFZU<1ORZU>6THEN1804
1807 ZU=ZU+6
1808 IFSB(ZU)=0THEN1804
1810 GOSUB2100
1860 AI=SB(IN):IFAI=2ORAI=4ORAI=6THEN1870
1865 GOTO1900
1870 HS=HS+AI:SB(IN)=0:IN=IN-1:IFIN=0THENIN=12
1880 GOSUB2000
1890 GOTO1860
1900 IFHS>36ORHC+HS=72THEN3000
1910 GOTO90
2000 PRINT"■■■■■     ";
2010 FORK=6TO1STEP-1:IFSB(K)<10THENPRINT" ";
2015 PRINTSB(K);:NEXT:PRINT
2020 PRINT:IFHC<10THENPRINT" ";
2022 PRINTHC;"                           ";:IFHS<10THENPRINT" ";
2024 PRINTHS
2030 PRINT:PRINT"     ";:FORK=7TO12:IFSB(K)<10THENPRINT" ";
2035 PRINTSB(K);:NEXT:PRINT
2036 PRINT"     ■■______________________________"
2037 PRINT"     ■ 1   2   3   4   5   6  "
2040 GOSUB4000:RETURN
2100 AN=SB(ZU):SB(ZU)=0
2110 FORJ=1TOAN:IN=ZU+J-INT((ZU+J)/12.01)*12:SB(IN)=SB(IN)+1
2120 GOSUB2000
2150 NEXT
2160 RETURN
3000 IFHC>HSTHENPRINT"■■■■■■■■■■■■■■■■■■■■■■■■■■■ICH HABE GEWONNEN!"
3010 IFHS>HCTHENPRINT"■■■■■■■■■■■■■■■■■■■■■■■■■■■SIE HABEN GEWONNEN!"
3020 IFHS=HCTHENPRINT"■■■■■■■■■■■■■■■■■■■■■■■■■■■UNENTSCHIEDEN!"
3030 INPUT"NOCH EIN SPIEL (J/N)";JN$:IFJN$="J"THENRUN
4000 FORI=15TO0STEP-5:POKE54296,I:POKE54276,129:POKE54277,15:POKE54273,40
4010 POKE57272,129:NEXT:POKE54276,0:POKE54277,0:RETURN

READY.
```

3/83 ist Erstausgabe

Ich habe die dritte Ausgabe ihrer Zeitschrift „Homecomputer" gelesen und war von dieser so beeindruckt, daß Ich nun die zwei ersten Ausgaben nachbestellen will. Schicken Sie mir bitte diese Hefte zu, wenn möglich mit Zahlkarte.

Jürgen Richter Bad Soden

Viele Anrufe und Zuschriften erreichten uns, in denen wir um Zusendung von Heft 1 und 2 gebeten wurden. Heft 3 - März 1983 ist die erste Ausgabe von Homecomputer gewesen. Danke für das Kompliment.

Zeichengenerator

Eine Meinung zu Ihrer Zeitung „Homecomputer". Die vielen Programme finde ich sehr gut, nur sind die Erläuterungen ziemlich spärlich. Besonders für das Programm aus Heft 3 Seite 42 – VC-261/HC – Der Zeichen-Generator. – Das Programm habe ich abgetippt, aber kann so recht nichts mit Anfangen. Wie mache ich es, daß ich die erstellten Zeichen in anderen Programmen verwenden kann? Zu diesen Programm fehlt eine genaue Erläuterung mit Beispielen. Dann noch etwas zu dem Programm, zum – Zeichen Duplizieren –, daß funktioniert bei mir nicht. Der Computer zeigt nur, daß aus dem Zeichen Ø wird. Könnte da ein Fehler im gedruckten Listung vorliegen?

Klaus Strecker Hagen

Das Listung ist ohne Fehler. Die Schwierigkeiten lassen sich sicher mit dem Artikel dann in dieser Ausgabe beseitigen.

Speicherplatzprobleme

Vor einer Woche habe ich mir das 1. Exemplar Ihrer Zeitschrift zugelegt, wobei mir sofort die Vielfalt der abgedruckten Programme angenehm aufgefallen ist. Ebenfalls finde ich gut, daß sich Zweck- und Spielprogramm die Waage halten.

Und nun muß ich meinem Ärger einmal Luft machen. Zu Hause habe ich einen Sinclair ZX 81. Schon die Eingaberoutine war fehlerhaft. Zeile 40 wird nicht ohne den „LET" Befehl genommen. Außerdem frage ich mich was der Pfeil im Maschinenprogramm bedeutet. Nach verschiedener Interpretation dieses Zeichens und daraus resultierendem mehrmaligen Eintippen des Programmes kommt immer eine B/10 Meldung, nach einem darauf folgenden Tastendruck stürzt der Computer ab. Daraufhin ging ich zu meinem Freund und wir tippten ROBOT in seinen VC 20, Ergebnis: ERROR IN 310. Nach mehrmaliger Programmprüfung war kein Tippfehler zu finden. Auch SLALOM läuft nicht. Durch die POKE-Befehle ist der Programmspeicher gleich voll. Daraufhin die Frage: Stimmt eigentlich überhaupt eines der „Listings"? Wie dem auch sei, ich hoffe Sie werden im nächsten Heft die Berichtigungen der Programme abdrucken und in Zukunft mehr Sorgfalt auf die „Listings" verwenden.

Andreas Kiel Bad Münster

VC-20 Absturz wird in diesem Heft erklärt. Wenn's dann nicht läuft liegt's wahrscheinlich daran, daß Ihr Freund weniger als 8K Speichererweiterung hat, und er die Programme deshalb nicht (oder nur nach erheblichem Aufwand - Umsetzen aller Pokebefehle) verwenden kann. Dafür spricht übrigens, daß das Programm Robot nicht läuft!

ZX 81 - U-Boot Jagd-Fehler

Über die Ersterscheinung Ihrer Zeitschrift „Homecomputer" bin ich hoch erfreut. Ich finde, daß es eine gelungene Alternative zu den bisher auf dem Markt erschienenen Magazinen ist. Leider muß ich bemängeln, daß das auf Seite 27 erschiene Programm für den 1 K-ZX-81 „U-Boot-Jagd" in dieser Form nicht lauffähig ist. (Es erscheint nach eingeben von „Rem" immer „B/10"; dann bricht das System zusammen). Nachdem ich das Programm mehrere Male vergeblich abgeschrieben habe, bitte ich Sie, mir zu schreiben, wo der Fehler liegt.

Martin Riedmüller Spaichingen

Es handelt sich um ein älteres Programm, daß auf einem der ersten ZX mit altem ROM programmiert wurde. Für neuere Geräte gilt:
16586 29 21 62 33 205 29 21 239

Apple Zeichensatz-Eproms

Eine Frage zu Apple Steno (Heft 3). Wo kann ich die erforderlichen Materialien (Zeichensätze, Eproms, Schalter usw.) beziehen und was Kosten diese?

Günter Britz Beckingen

Eproms mit Zeichensätzen erhalten Sie zum Preis von je 25,- DM über den Leserservice. Schalter gibt's im Fachhandel.

Dive Bomber

In Ihrem Programmen steht in Zeile 580 ein kleiner Fehler, der sicherlich Ti-Anfänger zur Verzweifelung treiben wird. Es muß an dieser Stelle **BY** = FY heißen und nicht **BX** = FY.

H. Hanswillemenke Köln

Leserbriefe

Schriftzeichen

Ihre Zeitschrift habe ich vor ein paar Tagen am Zeitungsstand eines Verbrauchermarktes entdeckt. Beim ersten Durchblättern störten mich allerdings die merkwürdigen Schriftzeichen etwas, mit denen sämtliche redaktionellen Seiten bedruckt sind. Vielleicht habe ich das Heft auch nur gekauft, um zuhause den Text in Ruhe entziffern zu können. Dabei habe ich dann festgestellt, daß ich das erste Exemplar eines neuen Computermagazins auf dem Tisch hatte.
Die Schrift nervte mich weiterhin. Diese „Computerschrift" ist nicht nur ausgesprochen häßlich, sondern auch noch hinderlich beim Lesen. Bestimmte Zeichen sind nur zu erraten. Wenn ich einen Artikel in einer Fachzeitschrift lese, will ich mich nicht an der Grafik der Buchstaben ergötzen, sondern will mich auf den Inhalt konzentrieren. An letzterem habe ich übrigens nichts zu nörgeln.
Gut finde ich den Artikel über den „Dornröschen-Schlaf". Ich befürchte aber, daß es bereits doch schon zu spät ist, aufzuwachen. Die abgedruckten Programme konnte ich noch nicht alle ausprobieren. Das Abtippen ist doch sehr mühsam. Aber dieses Problem kann ja durch Ihren Software-Service gelöst werden. Etwas Vergleichbares konnte ich noch in keiner Computerzeitschrift finden.
Bleibt noch zu bemerken, daß es nun auch ein Computer-Magazin gibt, in dem man keine großformatigen Farbfotos von Klein- und Großvieh mit seitenlangen Berichten über Mikrocomputer im Kuhstall findet. Welcher Heimanwender hat schon eine Rinderherde oder eine Schweinezucht hinter dem Haus? Ihren Vorsatz, die Leser mit „philosophischen Betrachtungen" und ähnlichen Dingen zu verschonen kann ich nur begrüßen. Ich wünsche Ihnen viel Erfolg und alles Gute für die Zunkunft Ihrer Zeitschrift.

Rainer Bärwolf — Gelsenkirchen

ATARI nicht vergessen

Per Zufall wurde ich auf Ihr Magazin aufmerksam. Ich kenne so ziemlich alle auf dem deutschen Markt erscheinenden Computermagazine von CHIP angefangen bis hin zu Computer persönlich.
Nach kritischer Durchsicht Ihres Magazines stelle ich fest, daß sich Homecomputer deutlich von den anderen Magazinen abhebt, und zwar im positivem Sinn, bedingt durch die vielen Listings für gängige Computersysteme. Nur für den ATARI habe ich leider nichts entdecken können, obwohl dieser Computer ein breites Feld von Möglichkeiten anbietet (leider nur kein Fußball-Bundesliga-Programm). Aber vielleicht ringt sich Ihr Mitarbeiterstab dazu durch, ihn nicht in der Versenkung verschwinden zu lassen. Einzige Kritik die ich äußern darf, ist die etwas ungewöhnliche Schriftform Ihres Magazines. Nicht gerade eine Augenweite, aber vielleicht gewöhnt man sich mit der Zeit daran.

Werner May — Krefeld

Wir sind gerade dabei, uns in das Atari System besser einzuarbeiten. Das wird noch einige Wochen dauern. Aber spätestens ab Ende April werden monatlich mehrere Atariprogramme vorliegen, die dann auch in Homecomputer abgedruckt werden.

... gleich eine erste Kritik: Die Schrift ... relativ schwer zu lesen, warum nicht normal (muß man umbedingt um jeden Preis „Originell" sein?)

Reinhard Schindler — Erlangen

Ich habe die Zeitschrift in einem Kiosk entdeckt. Da ich den VC 64 besitze und als Computerneuling an Software-Mangel leide, habe ich sofort zugegriffen. Im großen und ganzen ist Ihr „Angebot" gut. Aber einige Dinge ärgerten mich.
Lesbarkeit: Bitte wählen Sie doch Schrifttypen, die besser lesbar sind. Vielleicht sollte sich jemand bei Ihnen einmal mit Typografie beschäftigen. Aber abgesehen von der (UN) Ästhetik sind manche Zeilen einfach unleserlich.
Z. B. S. 10 Programmzeile 20 = ⎤ unlesbares
S. 11 Programmzeile 320 = ⎦ Steuerzeichen
S. 13 viele Steuerzeichen unlesbar
Besser sind die Schrifttypen auf S. 19 beim Programm: „Concorde„

Herbert Peschke — Gäufelden 1

... habe ich für Ihre Zeitschrift noch einige Änderungsvorschläge:
1. Schriftbild: Ich bin mit einigen Freuden der Meinung, daß ein anderer Zeichensatz – ohne „Balken" unten links an den Buchstaben – wesentlich besser lesbar ist.
2. Umlaute: Es dürfte meines Erachtens keine all zu großen Schwierigkeiten bereiten, den Schriftsatz um Umlaute und das „ß" zu erweitern.

Christoph Sauer — München 19

Tja, das Schriftbild ... Umlaute und „ß" sind allerdings vorhanden!

Sehr geehrte Homecomputer-Redaktion!
1. Zum Konzept Ihrer Zeitschrift: Erstklassig, viele sehr gute Programme für wenig Geld!
2. Die Möglichkeit die Programme auf Cassette geliefert zu bekommen und zu diesem Preis: Einfach Super!
3. Als ich die Inserate der englischen Softwarenhändler sah, wurde mir erst bewußt, wie rückständig in Deutschland die Softwareanbieter sind. Ich hoffe, daß durch Ihre Zeitschrift den Softwareanbietern in Deutschland die Augen etwas geöffnet werden.
4. Betr. ZX81 – U-Boot Jagd. Das war bisher meine einzige Enttäuschung in dem Heft.

Michael Birkle — Tübingen

„der 16-bittige“ = 68 000 cpu comp.

8 MHz takt · 64 + 128 KRAM = 192
komfort. editor/assembler
tastatur, 12“-monitor, nur
5000.– dm
(softw. + sprachen anfr.–
teilw. apple kompatibel)

„das grüne“ = recycling-edv-papier

240X12“ (A4) lochr. perf. 2000 blatt **35.- dm**
weitere ausführungen,
formate, formulare anfragen

diestel gmbh aachen problemlösungen

abt. mycros
kaiserstr. 134
5102 würselen
preise inkl. mwst.
zuzügl. porto + nn
(ups)

„top“ = monitor + disk 2 X 5¼“

in **einem** Gehäuse f. apple u. ä.
m. Contr. **DM 2250.-**
(endlich keine
„Turmbauten“ mehr)

„die 16-bittige“ = 68 000 cpu f. apple

8 MHz Takt + 128 KB RAM + Ed/Ass
und Pseudodisk **DM 1977.50**
FORTH, BASIC,
PASCAL anfr.

ca. 50 versch. karten f. apple u. ä. fragen sie uns!

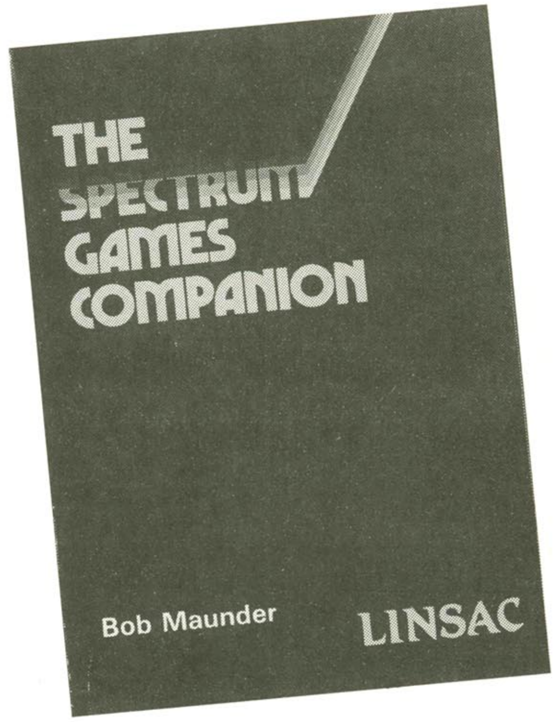
Ekkehard Flögel
Programmieren in Basic und Maschinencode mit dem ZX81



128 S. 29.80 DM
Hofacker Holzkirchen

Ideal geeignet für Einsteiger, die sich über die Möglichkeiten ihres ZX-81-Rechners informieren wollen und vollständige Programmlistings und konkrete Tips erwarten.
In Kapitel "Spiele" wird Schritt für Schritt vorgeführt, wie einfache Actionspiele entstehen können, vom Entwerfen von Figuren über Schlangen, die Zeichen fressen bis zu Enten, die abgeschossen werden sollen. Insbesondere kann man sich hier mit dem geschickten Umgang mit der INKEY$-Funktion (Tastaturabfrage) und der RND-Funktion (Zufallszahlen) vertraut machen. Auch der fortgeschrittene Anwender wird noch auf kleine Programmiertricks stoßen, die seine Programme um ein Stück eleganter machen können.
Ein Kapitel "Programme für die Schule" stellt unter anderen Algorithmen für die Primfaktorzerlegung und die Lösung quadratischer Gleichungen vor. Ebenso werden grafische Darstellung von Funktionen und der Rechner als Einpauker von Vokabeln behandelt.
Ein Schallplattenverzeichnis und ein Terminkalender sind umfangreiche Programme aus dem Kapitel "Datenverwaltungs".
Wer sich mit Maschinencode und darauf aufbauend mit der Aufnahme von Daten aus der Umwelt und mit Steuerungsaufgaben befassen will, bekommt in drei weiteren Kapiteln die Gelegenheit, erste Maschinenprogramme zum Laufen zu bringen.
Als sehr nützlich (auch für den Fortgeschrittenen) erweist sich eine alphabetische Auslistung des Befehlssatzes des Z80A-Microprozessors mit knapper Funktionsangabe im Anhang.

Bob Maunder: The Spectrum Games Companion



Middlesbrough Linsac 1982
122 S. 5,95 engl. Pfund
ISBN 0 907211 02 X

In einem Eingangskapitel gibt Maunder Kriterien an, mit denen seiner Meinung nach die Qualität von Spielprogrammen gemessen werden kann und zeigt einen methodischen Weg auf, in diesem Sinne gute Software zu entwickeln.
In den folgenden 7 Kapiteln sind jeweils 3 Spiele des gleichen Charakters in einem Kapitel zusammengestellt. Es handelt sich um Zahlen-, Wort-, Brett-, Simulations-, Würfel-, Karten- und Such- und Fangspiele.
Jedes Spiel wird vorgestellt mit einer globalen Beschreibung, einer systematischen Aufstellung der einzelnen Programmschritte mit Verweisen auf die Programmzeilennummern, einem typischen Schirmbild, einer Variablenliste mit Angabe der Bedeutungen, dem Programmlisting und abschließenden Hinweisen auf besondere Programmiermethoden.
Es sind durchweg reizvolle Spiele. Einige waren in ihrer Grundstruktur bereits im Begleitbuch zum ZX-81 zu finden. Auch Spielen, die einer großen Zahl von Lesern bekannt sein dürften, gewinnt Maunder durch eine kleine Regeländerung einen neuen Aspekt ab (Solitaire, Mastermind). Er schafft es, auch die Wortspiele durch begleitende Grafik interessant und spannend zu gestalten.
Dem bisherigen ZX-Benutzer bietet sich in den Programmlistings die Gelegenheit, die neuen reichhaltigen Möglichkeiten des Spectrum kennenzulernen.

Bandwurm für EG2000 Video-Genie

Bei diesem Programm handelt es sich um ein interessantes und unterhaltsames Taktikspiel, das viele Computerbesitzer ansprechen wird.
Der Spieler hat die Aufgabe, ein Quadrat mittels der vier Cursortasten über den Bildschirm zu steuern. Dabei muß er aber einige Dinge beachten. Er darf sein Quadrat immer nur auf leere Bildschirmstellen steuern. Da aber das Quadrat wandert und die jeweilige alte Quadratposition nicht gelöscht wird, füllt sich der Bildschirm langsam mit einer Kette von Quadraten. Der BANDWURM entsteht.
Um die Problemstellung reizvoller zu gestalten, sind in das Programm noch einige Kniffe eingebaut.
Damit der Spieler nicht in aller Ruhe seinen Weg suchen kann, muß er sein Quadrat immer in Bewegung halten. Die Zeitspanne variiert je nach gewähltem Schwierigkeitsgrad zwischen 3 bis 1/3 Sekunde.
Da er, wie schon erwähnt, seinen Bandwurm immer nur auf freie Positionen lenken darf, muß der Spieler ihn auch von der Spielumrandung fernhalten. Läßt er ihn jedoch mit der Umrandung kollidieren, so stirbt sein Bandwurm. Damit sich der Spieler nicht schon nach kurzer Zeit eingemauert hat, löscht das Programm von Zeit zu Zeit einzelne Bandwurmquadrate.
Durch die so entstehenden Lücken kann der Spieler seinen Bandwurm steuern, um aus einem verschlossenen in ein offenes Labyrinth zu gelangen.
Die eigentliche Aufgabe aber ist es, Pluspunkte zu sammeln. Dies geschieht durch die Bewegung des Bandwurms. Für jeden Schritt, den der Bandwurm zurücklegt, erhält der Spieler einen Punkt. Den momentanen Punktstand kann er am unteren Bildrand ablesen. Periodisch erscheinen auf dem Bildschirm MIKROBEN. Diese kleinen Mikroben sind für den Bandwurm eine besondere Delikatesse. Wenn es dem Spieler möglich ist, seinen Bandwurm auf eine solche Mikrobe (kleiner Punkt) zu steuern, so erhält er 100 Punkte.

Uwe Michalski

UMSETZTABELLE FÜR GRAPHISCHE ZEICHEN UND SONDERZEICHEN

Kleinbuchstabe	ASCII	graphisches Zeichen
a	64	
b	202	
c	209	
d	213	
e	212	
f	210	
g	246	
h	217	
i	255	
j	253	
k	223	
m	211	
n	227	
o	207	
p	244	
q	219	
r	238	
s	245	
t	229	
u	218	
v	251	
w	232	
ü	62	> Größer als
ä	60	< Kleiner als
§	42	* Multiplikationszeichen
.	32	Blank Leerzeichen



Alle Kleinbuchstaben (graphische Zeichen & Sonderzeichen) können direkt über die Tastatur eingegeben werden.

Fremdsprache

Bei diesem Programm handelt es sich um ein Vokabellernprogramm.

In dieses Programmskelett muß der Anwender die Vokabeln der Sprache, die er erlernen will, einfügen.

Das Programm fragt diese dann der Reihe nach oder auch unregelmäßig durcheinander ab.
Das Programm ist so einfach gehalten, daß es ohne Schwierigkeiten mit kleinen Änderungen auf jeden Heimcomputer zum Ablauf gebracht werden kann.

Uwe Michalski

Das Vokabellern-Programm befindet sich auf Seite 16

```
1REM
2REM------ B A N D W U R M ------
3REM
4REM-- ENTWICKELT BY DOMI-TEAM --
5REM
6REM--          3.1.83          --
7REM
8REM-- ALLE RECHTE VORBEHALTEN --
9REM
1ØCLS
11COLOUR4
12GOSUB48
13P=5ØØ
14COLOUR3
15PRINTaØ,"bccccccccccccccccccccccccccccccccccccccb"
16FORX=4ØTO84ØSTEP4Ø:PRINTaX,"d......................................e";
  :NEXT:PRINT"bffffffffffffffffffffffffffffffffffffffb";
17COLOUR1:PRINTa929,"PUNKTE: ";
18COLOUR6
19PRINTaP,"b"
2ØC=PEEK(-1984)
21IFCAND64THENP=P+1:IFPEEK(174Ø8+P)=32THEN29ELSEGOTO37
22IFCAND32THENP=P-1:IFPEEK(174Ø8+P)=32THEN29ELSEGOTO37
23IFCAND16THENP=P+4Ø:IFPEEK(174Ø8+P)=32THEN29ELSEGOTO37
24IFCAND8THENP=P-4Ø:IFPEEK(174Ø8+P)=32THEN29ELSEGOTO37
25S=S+1
26IFSüBTHEN37
27IFFä5ØTHENF=F+1ELSEU=(RND(37)+1)+((RND(2Ø)+1)§4Ø):COLOUR2:PRINTaU,"g"
  :F=Ø:GOTO2Ø
28GOTO2Ø
29COLOUR6:PRINTaP,"b"
3ØZ=(RND(37)+1)+((RND(2Ø)+1)§4Ø):PRINTaZ,"."
31L=L+1
32COLOUR1
33PRINTa938,L+V;
34PLAY(1,5,6,11):PLAY(1,5,6,Ø)
35S=Ø
36GOTO27
37REM-- BIG BANG --
38IFPEEK(174Ø8+P)=ASC("g")THENV=V+1ØØ:PLAY(1,3,3,1Ø):FORQ=1TO3Ø:NEXT
  :PLAY(1,3,3,Ø):GOTO29
39COLOUR8
4ØPLAY(3,4,7,11):FORX=1TO3ØØ:NEXT:PLAY(3,4,7,Ø)
41CLS:FORX=1TO2ØØ:BGRD:NBGRD:NEXT
42PRINTa4ØØ," DER BANDWURM HAT FUER DICH INSGESAMMT "
43PRINTa49Ø,V+L;:PRINT" PUNKTE ERREICHT"
44PRINTa8ØØ," hhhi WILLST DU NOCHMAL SPIELEN ? jhhh"
45PRINTa88Ø,"..............JA ODER NEIN"
46A$=INKEY$:IFA$=""THEN46ELSEIFA$="J"THENCLS:GOSUB73:L=Ø:V=Ø:GOTO13ELSE
  IFA$="N"THENENDELSEGOTO46
47END
48FORH1=1TO8:FORH2=1TO12:PLAY(1,H1,H2,13):NEXT:NEXT:PRINT"lmmmmmmmmmm
  mmmmmmmmmmmmmmmmmmmmmmmmmmmmmmn";
49PRINT"o........pmmmmmmmmmmmmmmmmmmmp.........q";
5ØPRINT"o.......rbs";:COLOUR5:PRINT" B A N D W U R M";:COLOUR4:PRINT".rbs
  ........q";
51PRINT"o........tuuuuuuuuuuuuuuuuuuut.........q";
52PRINT"o......................................q";
53PRINT"o..DU MUSST DEN BANDWURM MIT DEN VIER..q";
54PRINT"o..PFEILTASTEN (i.j.v.w) IMMER IN......q";
55PRINT"o..BEWEGUNG HALTEN! WENN DER WURM ZU...q";
56PRINT"o..LANGE AN EINER STELLE BLEIBT STIRBT.q";
57PRINT"o..ER AUS BEWEGUNGSMANGEL. ALS NAHRUNG.q";
58PRINT"o..VERZEHRT ER MIKROBEN ü ";:COLOUR2:PRINT"g";:COLOUR4:PRINT".ä
  .!........q";
59PRINT"o..WENN ER ALLERDINGS SEINE EIGENEN....q";
6ØPRINT"o..BANDWURMGLIEDER FRESSEN MUSS, DANN..q";
61PRINT"o..GEHT ER EBENFALLS EIN!..............q";
62PRINT"o..JE LAENGER ER LAEUFT UND MIKROBEN...q";
63PRINT"o..VERZEHRT, UMSO HOEHER IST DER.......q";
64PRINT"o..PUNKTESTAND DEN ER FUER DICH SAMMELTq";
65PRINT"o..ALSO STEUERE IHN GUT!...............q";
66PRINT"o..PS. WENN DU DEN WURM MIT DER FELD-..q";
67PRINT"o......UMRANDUNG KOLLIDIEREN LAESST....q";
68PRINT"o......DANN STIRBT ER AUS PROTEST!.....q";
69PRINT"o...BITTE üRETURNä DRUECKEN............q";
7ØPRINT"nuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuul";
71FORH1=8TO1STEP-1:FORH2=12TO1STEP-1:PLAY(1,H1,H2,13):NEXT:NEXT:PLAY
  (1,1,1,Ø)
72I$=INKEY$:IFI$=""THEN72ELSEIFASC(I$)=13THEN73ELSEEND
73COLOUR4:CLS
74PRINT"lmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmn";
75PRINT"o........pmmmmmmmmmmmmmmmmmmmp.........q";
76PRINT"o.......rbs";:COLOUR5:PRINT" BANDWURM";:COLOUR4:PRINT".rbs
  ........q";
77PRINT"o........tuuuuuuuuuuuuuuuuuuut.........q";
78PRINT"o......................................q";
79PRINT"o..NUN MUSST DU NOCH DEINEN SPIELGRAD..q";
8ØPRINT"o..FESTLEGEN. ES STEHEN DIE FOLGENDEN..q";
81PRINT"o..SCHWIERIGKEITSGRADE ZUR AUSWAHL:....q";
82PRINT"o......................................q";
83PRINT"o..GRAD:   1. . . .  PROFI.............q";
84PRINT"o............v.........................q";
85PRINT"o...........9. . . .  ANFAENGER........q";
86PRINT"o......................................q";
87PRINT"o..BEI SCHWIERIGKEITSGRAD 9 KANNST DU..q";
88PRINT"o..DEN BANDWURM ETWA DREI SEKUNDEN LANGq";
89PRINT"o..AN EINER STELLE STEHEN LASSEN OHNE..q";
9ØPRINT"o..DASS ER STIRBT. BEI STUFE 1 NUR ETWAq";
91PRINT"o..1/3 SEKUNDE LANG!...................q";
92PRINT"o..WAEHREND DIESER ZEITSPANNE KANNST DUq";
93PRINT"o..DEN WEG ZU DER NAECHSTEN MIKROBE....q";
94PRINT"o..SUCHEN!.............................q";
95PRINT"o.hi.GIB DEINEN SOIELGRAD EIN (1-9) jh.q";
96PRINT"nuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuul";
97T$=INKEY$:IFT$=""THEN97ELSEIFT$="Ø"THENENDELSEIFTSä"1"ORTSü"9"THENCLS:
  GOTO73ELSET=VAL(T$):B=INT(T)§5:S=Ø:RETURN
```

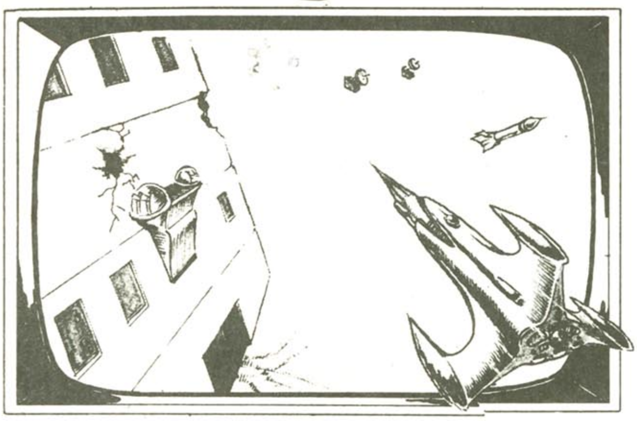
```
    1 LET R=0
    2 DIM A$(5000)
    3 DIM B$(5000)
    4 DIM C(50)
    5 E=0
    6 REM ****     BEGINN DER DATEN BEI 10  ZUERST A$ (FREMDSPRACHE)
    7 REM ****     Z.B.    10 A$(1)="DOG"   11 B$(1)="HUND"
11000 REM *** BEGINN DES PROGRAMMS ***
11009 PRINT "*****************************"
11010 PRINT "*** XXXX. - DEUTSCH ===>1 ***"
11015 PRINT "*** DEUTSCH - XXXX. ===>2 ***"
11016 PRINT "*** COMPUTERWAHL     ===>3 ***"
11017 PRINT "*****************************"
11020 INPUT S
11030 IF S=3 THEN 15000
11040 PRINT "============================="
11045 PRINT "WIEVIEL VOKABELN?"
11050 INPUT Z
11055 PRINT " VON WELCHER STELLE?"
11060 PRINT " BITTE ZAHL EINGEBEN"
11065 INPUT V
11070 IF S=2 THEN 11300
11075 FOR K=V TO V+Z-1
11080 PRINT "DEUTSCHE FORM VON ";A$(K);" ? "
11085 INPUT L$
11090 IF L$="STOP" THEN 20000
11095 IF L$=B$(K) THEN 11200
11100 PRINT " FALSCH !!!!! "
11105 PRINT " RICHTIG WAERE ";B$(K)
11110 GOTO 11250
11200 PRINT " RICHTIG "
11205 LET R=R+1
11250 NEXT K
11255 GOTO 11910
11300 REM *** DEUTSCH -FREMDSPRACHE.-TEST ***
11302 FOR J=V TO Z+V-1
11305 PRINT " XXXXXXXXX FORM VON ";B$(J);" ? "
11310 INPUT O$
11315 IF O$="STOP" THEN 20000
11320 IF O$=A$(J) THEN 11350
11325 PRINT " FALSCH !!! "
11330 PRINT " RICHTIG WAERE ";A$(J)
11335 GOTO 11400
11350 PRINT " RICHTIG  "
11355 LET R=R+1
11400 NEXT J
11900 REM *** ANTWORTEN ZAEHLEN ***
11905 LET W=Z-V+1
11910 PRINT " ABGEFRAGTE VOKABELN ";Z
11915 PRINT " VON STELLE ";V;" BIS ";V+Z
11920 PRINT " RICHTIGE ANTWORTEN ";R
11925 PRINT " FALSCHE ANTWORTEN ";Z-R
11930 GOTO 20000
15000 PRINT "**************************"
15005 PRINT "**** XXX.-DEU.===> 1   ****"
15010 PRINT "**** DEU.-XXX.===> 2   ****"
15015 PRINT "**************************"
15020 INPUT P
15030 REM ******** XXX.-DEU.-TEST **********
15035 PRINT " WIEVIEL DEU. VOKABELN SIND VORHANDEN ? "
15040 INPUT U
15043 IF P=2 THEN 15500
15045 FOR X=1 TO 20
15050 C(X)=INT(U*RND(-1)+1)
15055 NEXT X
15060 FOR Z=1 TO 20
15065 PRINT " DEUTSCHE  FORM VON ";A$(C(Z))
15070 INPUT F$
15075 IF F$="STOP" THEN 20000
15080 IF F$=B$(C(Z)) THEN 15100
15085 PRINT " FALSCH "
15090 PRINT " RICHTIG WAERE ";B$(C(Z))
15095 GOTO 15120
15100 PRINT " RICHTIG "
15105 LET E=E+1
15120 NEXT Z
15125 GOTO 16000
15500 REM ***** DEU./ XXX. TEST ****
15505 FOR I=1 TO 20
15510 C(I)=INT(U*RND(-1)+1)
15515 NEXT I
15520 FOR W=1 TO 20
15525 PRINT " XXXXXXXXX FORM VON ";B$(C(W))
15530 INPUT D$
15535 IF D$="STOP" THEN 20000
15540 IF D$=A$(C(W)) THEN 15560
15545 PRINT " FALSCH "
15550 PRINT " RICHTIG WAERE ";A$(C(W))
15555 GOTO 15570
15560 PRINT " RICHTIG "
15565 LET E=E+1
15570 NEXT W
15575 GOTO 16000
16000 PRINT "@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@"
16003 PRINT "@@ 20 ABGEFRAGTE VOKABELN     @@"
16005 PRINT "@@ DAVON RICHTIG ";E;"        @@"
16310 PRINT "@@ DAVON FALSCH ";20-E;"      @@"
16312 PRINT "@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@"
16315 GOTO 20000
19000 REM
20000 END
```

Anpassungshinweise

Zeile 2+3: Die DIM-Anweisungen entsprechend den Speicherverhältnissen anpassen.

Zeile 15510+ 15050: C(x)= Zufallszahl zwischen 1-U
C(X)=INT(RND(U)) / C(I)=INT(RND(U))
- EG2000_Format -

Tonight On Your Micro



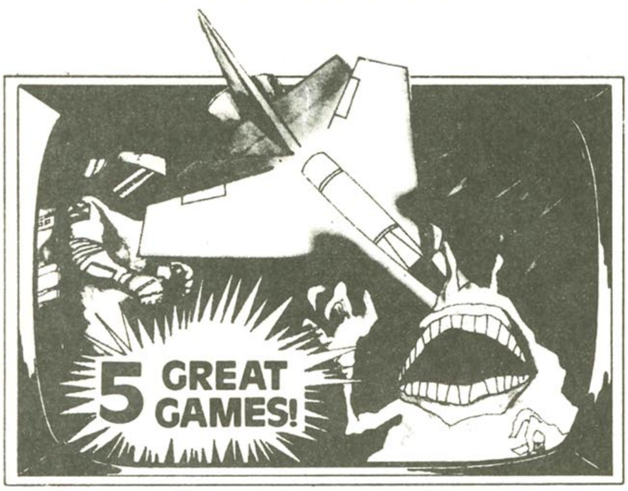
CAN YOU SURVIVE PENETRATOR?

Penetrator is the most amazing and sophisticated arcade game yet devised for the 48K Spectrum with features never seen before! Cassette £6.95.



RESCUE A VIC PRINCESS

A multi-part adventure for the standard VIC 20! All you have to do is find the castle, defeat the dragon, find your way through the labyrinth, kill the wizard and rescue the Princess. And that's not half of it! The Wizard & The Princess cassette only £5.95.



CONVERT YOUR VIC INTO AN ARCADE MACHINE

Amazing arcade action for the standard VIC 20 with 5 new exciting games from Clifford Ramshaw: ALIEN BLITZ* INVADERS* GROUND ATTACK, STORM and SPACE ROCKS* (games marked * contain machine code). You must have the GAMES PACK cassette–only $5.95.



TALK TO THE SPECTRUM!

Why is this man smiling? You will too when you read Dr. Logan's book UNDERSTANDING YOUR SPECTRUM. The definitive guide for only £7.95.

MELBOURNE HOUSE PUBLISHERS

Orders to: 131 Trafalgar Road, Greenwich, London SE10

Correspondence to: Glebe Cottage, Station Road, Cheddington, Leighton Buzzard, BEDS LU7 7NA

HC 2

☐ Please send me your free catalogue.
Please send me:

Cassettes

16K SPECTRUM

☐ Over The Spectrum No. 1	£5.95
☐ Over The Spectrum No. 2	£5.95
☐ Over The Spectrum No. 3	£5.95
☐ Programs from Spectrum Machine Language Book	£5.95

48K SPECTRUM

☐ Penetrator	£6.95
☐ The Hobbit	£14.95

STANDARD VIC 20

☐ VIC Innovative Cassette 1	£5.95
☐ VIC Innovative Cassette 2	£5.95
☐ VIC Innovative Cassette 3	£5.95
☐ VIC Games Pack	£5.95
☐ The Wizard & The Princess	£5.95

Books

SPECTRUM

☐ Understanding Your Spectrum	£7.95
☐ Spectrum Machine Language For The Absolute Beginner	£6.95
☐ Over The Spectrum	£6.95

VIC 20

☐ VIC Innovative Computing	£6.95

All prices include VAT where applicable

Unconditional Guarantee All Melbourne House cassette software is supplied on high quality cassettes with attractive inserts. Full instructions are provided and the cassettes are unconditionally guaranteed against malfunction.

£ ________

Please add 80p for post and pack £ .80

TOTAL £ ________

I enclose my cheque/money order for £ ________

Please debit my Access card No. ________ Expiry date ________

Signature ________ Name ________

Address ________

________ Postcode ________

MELBOURNE HOUSE PUBLISHERS



Step into the future:

VISIT MIDDLE EARTH WITH YOUR SPECTRUM.

Melbourne House Publishers, in association with the Tolkien Estate, are very proud to announce the release of THE HOBBIT, a super-program like no other before, for the 48K Spectrum.

THE HOBBIT is totally unique, with features never before seen on any computer, let alone on the Spectrum.

Based on the fantasy land described in J. R. R. Tolkien's brilliant novel, in this program you take on the role of Bilbo, the hobbit: danger, adventure, and excitement are all part of it, presented to you in words and graphics, but it is you who must confront and solve the problems this time.

You instruct the computer in completely ordinary English sentences!

THE HOBBIT program is capable of very sophisticated communications, including features not previously thought possible on a microcomputer: adjectives, adverbs, multiple sentences, are just a part of the power of this program.

You will meet all your favourite characters from J. R. R. Tolkien's book, and amazing as it sounds, they will all have an independent life of their own. They can make decisions and act on them! They react not only to what you do, but also to every other character in THE HOBBIT!

Because of this unique feature, you will find that each time you play THE HOBBIT, events will proceed in a slightly different way, and the further you get into Tolkien's world, the more different each visit may be.

THE HOBBIT also allows you to talk to the other characters! At last you can answer the questions they may put to you, or more importantly, you can suggest to them what you would like them to do!

THE HOBBIT program follows closely the plot of the book, and so a copy of the book is included as part of THE HOBBIT package at no extra charge.

THE HOBBIT brings to you the future of microcomputers in an exciting and challenging fantasy. A super-program that is a milestone in computer software.

Other cassette software includes PENETRATOR, a 48K state-of-the- art arcade game–£6.95 including VAT.
Spectrum books include: Over The Spectrum–£6.95.
Spectrum Machine Language–£6.95.
Understanding Your Spectrum–£7.95.

Melbourne House Publishers HC 2
Orders to: 131 Trafalgar Road, Greenwich, London SE10.
Correspondence to: Glebe Cottage, Station Road, Cheddington, Leighton Buzzard, BEDS LU7 7NA.

☐ Please send me your **free** Spectrum Catalogue.

Please send me:

☐ The Hobbit package–48K Spectrum cassette, full colour instruction manual and Hobbit book–£14.95

☐ Other ....

(Please add 80p for post & pack.) I enclose cheque/money order £.............. for my Access card No.......................... Expiry date..........

Signature..

Name ....

Address ....

....

.... Postcode.....................

MELBOURNE HOUSE PUBLISHERS

Suchrätsel - selbst gemacht

In vielen Zeitschriften und Rätselheften findet man eine reizvolle Sorte von Rätseln - das Suchrätsel.
Ein Computer ist geradezu prädestiniert, diese Art von Rätseln herzustellen.
Homecomputer, die rätselhafte Zeitschrift, stellt ein Programm zur Erzeugung von Suchrätseln vor.

Suchrätsel werden in verschiedenen Formen in den angesprochenen Publikationen abgedruckt. Sie alle haben gemeinsam, daß man Wörter, die waagrecht, senkrecht oder diagonal in allen Richtungen auf dem Papier verteilt wurden, finden muß und sie dann einrahmt.
Die Worte, die gefunden werden sollen können vorgegeben sein oder sind zur Erschwernis der Sache nicht neben dem Rätsel abgedruckt.
Meist geht es in letzteren Falle um ein bestimmtes Thema, zu dem man entsprechende Begiffe suchen soll.
Unser Beispiel dreht sich, wie könnte es anders sein - um Homecomputer.
Das Programm fragt am Anfang nach der Größe des Rätsels. Diese kann auf den jeweiligen Drucker abgestimmt werden.
Danach wird um Eingabe der zu suchenden Worte gebeten.
Ist dieser Teil beendet, so wird zunächst die Lösung gedruckt, dann die Worte der Länge nach geordnet, und schließlich das eigentliche Rätsel, gefolgt von einem weiteren Abdruck der zu suchenden Begriffe.
Die Geschwindigkeit der Herstellung eines Rätsels richtet sich nach Größe des Feldes und Anzahl der Wörter. Es kann in seltenen Fällen vorkommen, daß das Programm in eine Endlosschleife gerät, weil der gerade zu verteilende Begriff nicht mehr in Feld untergebracht werden kann. Das erkennt man daran, daß sich über einen längeren Zeitraum nichts mehr tut. In einem solchen Fall sollte mit etwas weniger Worten oder mit einem etwas vergrösserten Feld neu begonnen werden.

```
M G X E M M J S I H U T B E F O E A J S
L X K Q U T O P Y F T L J H H T M C F U
Y R H H A R I C N B F P M U I R T E I F
H E A E P P A C S T Z G X R P I P U L F
P T P W P Q Z Z S A K T K G I G F U I T
H U W L L I H T A B T A N D Y Q I M S Y
P P Q E E C E N R B E P S O N K B S R W
T M U T U O L D O C R C U I H E Z I I A
Y O V T M B D U Q G P V C T D E F Q Z R
G C I K Q A Q B M D A Y I U S T O Z T W
S E W S J S Y Q Z H W R M Q B Y O Y V B
H M A W C I D A R N O C D C Y B T N K T
A O I T R S E J T Y F V U F Y G Y Z D Z
R H I H Y E E I M M R W U P F U F S C O
P O J A K V G R A R Q I R P I B E F N U
S X T Q B H E E O B V B A F M G B A F J
X V I K X U S Q A D J W S L V D O T W E
V M G R A P A E N R O B S O C A U A A B
X Q G T E R W C A A D M G V H N S R V W
P G K X F H H B K M S D M D F D I I W D
N V M R G O C E Z X X S B O L J C S J P
D D N X X V S S L X I J D D C T W E L T
F D A I M E D G T G J X N H U T L F O W
O F B F V E K M N R E K R D T P G J B L
C W C J I K K I L I A R O E B H W G W P
T E X A S N X G K U K B C M J E G I G N
V F N Z E W K B T X X C A L M D E H U H
A N W N E F P X R W L W G H O T I P M I
W T W S L G H C I R O Z K P R X Y X T R
J H G Y K X P O T R E M P H J V M K H B
```

Oktopus

Das Spiel Oktopus für den Apple II ist ein Reaktionsspiel, bei dem es auf Fingerfertigkeit und Schnelligkeit ankommt.

Jeder der beiden Spieler versucht, mit einem Paddle den schwarzen Punkt, den er durch sein Paddle steuern kann, in die gleiche Richtung zu drehen, aus der im großen Kreuz der wandernde Punkt erscheint.
Wer zuerst seinen Paddleknopf gedrückt hat, erhält einen Zähler gutgeschrieben. Bei 20 und wenn mindestens zwei Punkte Abstand zwischen den beiden Spielständen sind, ist ein Spiel beendet.

Die Programme für Suchrätsel-Erzeuger und Oktopus befinden sich auf den folgenden Seiten.

Die im Heft 3/83 angekündigte Sprite-Grafik wurde auf das folgende Heft 5/83 verschoben, weil noch einige interessante Funktionen, wie Background-Collison und Sprite-Sprite-Collision-Erkennung verbessert wurden.

```
ÜLIST

50  HOME
60  SPEED= 255
100  GR :A0 = 26:B0 = 5:A1 = A0:B1 = 35: POKE  - 16302,0
101 D0 = 30.652:D1 = 32:S =  - 16336
102 L1 =  - 16286:L0 =  - 16287
105  COLOR= 3
110  FOR I = 1 TO 47: HLIN 1,39 AT I: NEXT
111  COLOR= 0: VLIN 40,47 AT 0
115  COLOR= 15: VLIN 26,35 AT 5: HLIN 3,7 AT 30: HLIN 3,7 AT 31
116  VLIN 26,35 AT 35: HLIN 33,37 AT 30: HLIN 33,37 AT 31: VLIN 26,27 AT 3: VLIN 26,27 AT 7: VLI
N 26,27 AT 33: VLIN 26,27 AT 37
117  VLIN 34,35 AT 3: VLIN 34,35 AT 7: VLIN 34,35 AT 33: VLIN 34,35 AT 37
120  COLOR= 0: VLIN 4,37 AT 20
122  HLIN 12,28 AT 20: HLIN 12,28 AT 21
125 Z1 = 4:Z2 = 36: FOR I = 12 TO 28: PLOT I,Z1: PLOT I,Z1 + 1: PLOT I,Z2: PLOT I,Z2 + 1:Z1 = Z1
 + 2:Z2 = Z2 - 2: NEXT
150 A2 = 19:B2 = 20:R =  RND (1) * 8:R =  INT (R) + 1:ZZ = 0
155 F1 = 0:F0 = 0
157  FOR I = 1 TO 1000: NEXT
158  COLOR= 3: FOR I = 41 TO 47: HLIN 1,39 AT I: NEXT
160  ON R GOSUB 3100,3200,3300,3400,3500,3600,3700,3800
200  COLOR= 0: GOSUB 4000:ZZ = ZZ + 1
202  FOR B = 1 TO 5:BO =  PEEK (S) -  PEEK (S) +  PEEK (S) -  PEEK (S): NEXT
205  IF ZZ = 17 THEN  COLOR= 1: VLIN A2,A2 + 1 AT B2: GOTO 500
210 P0 =  INT ( PDL (0) / D0) + 1:P1 =  INT ( PDL (1) / D1) + 1
213 T1 =  PEEK (L1):T0 =  PEEK (L0)
215  COLOR= 15
220  ON P0 GOSUB 1100,1200,1300,1400,1500,1600,1700,1800
225  COLOR= 15
230  ON P1 GOSUB 2100,2200,2300,2400,2500,2600,2700,2800
235  IF T1 > 127 AND T0 > 127 AND F0 = 0 AND F1 = 0 AND R = P1 AND R = P0 THEN PU(0) = PU(0) + 1
:PU(1) = PU(1) + 1:RF(0) = 1:RF(1) = 1: GOTO 500
240  IF T1 > 127 AND F1 = 0 THEN 300
250  IF T0 > 127 AND F0 = 0 THEN 400
260  GOTO 200
300  IF R = P1 THEN PU(1) = PU(1) + 1:RF(1) = 1: GOTO 500
310 F1 = 1
320  GOTO 250
400  IF R = P0 THEN PU(0) = PU(0) + 1:RF(0) = 1: GOTO 500
410 F0 = 1: GOTO 200
500  COLOR= 0
510  FOR I = 42 TO 46: IF RF(0) = 1 THEN  HLIN 2,19 AT I
520  IF RF(1) = 1 THEN  HLIN 21,38 AT I
530  NEXT
540 X% = 2:Y% = 7:Z$ =  STR$ (PU(0)): IF PU(0) < 10 THEN Z% = 0: GOSUB 5000: GOTO 550
545 Z% =  VAL ( LEFT$ (Z$,1)): GOSUB 5000
550 X% = 6:Z% =  VAL ( RIGHT$ (Z$,1)): GOSUB 5000
560 X% = 32:Z$ =  STR$ (PU(1)): IF PU(1) < 10 THEN Z% = 0: GOSUB 5000: GOTO 570
565 Z% =  VAL ( LEFT$ (Z$,1)): GOSUB 5000
570 X% = 36:Z% =  VAL ( RIGHT$ (Z$,1)): GOSUB 5000
580 RF(0) = 0:RF(1) = 0
700  CALL 65338
710  IF  ABS (PU(0) - PU(1)) < 2 THEN 115
720  IF PU(0) < 20 AND PU(1) < 20 THEN 115
730  POKE  - 16301,0: HOME
740  VTAB 22: HTAB 10: PRINT "NOCH EIN SPIEL ? ";: GET A$: IF A$ = "J" THEN  RUN
750  END
1100  VLIN A0,A0 + 1 AT B0: COLOR= 0: VLIN 26,27 AT 5:A0 = 26:B0 = 5: RETURN
1200  VLIN A0,A0 + 1 AT B0: COLOR= 0: VLIN 26,27 AT 7:A0 = 26:B0 = 7: RETURN
1300  VLIN A0,A0 + 1 AT B0: COLOR= 0: VLIN 30,31 AT 7:A0 = 30:B0 = 7: RETURN
1400  VLIN A0,A0 + 1 AT B0: COLOR= 0: VLIN 34,35 AT 7:A0 = 34:B0 = 7: RETURN
1500  VLIN A0,A0 + 1 AT B0: COLOR= 0: VLIN 34,35 AT 5:A0 = 34:B0 = 5: RETURN
1600  VLIN A0,A0 + 1 AT B0: COLOR= 0: VLIN 34,35 AT 3:A0 = 34:B0 = 3: RETURN
1700  VLIN A0,A0 + 1 AT B0: COLOR= 0: VLIN 30,31 AT 3:A0 = 30:B0 = 3: RETURN
1800  VLIN A0,A0 + 1 AT B0: COLOR= 0: VLIN 26,27 AT 3:A0 = 26:B0 = 3: RETURN
2100  VLIN A1,A1 + 1 AT B1: COLOR= 0: VLIN 26,27 AT 35:A1 = 26:B1 = 35: RETURN
2200  VLIN A1,A1 + 1 AT B1: COLOR= 0: VLIN 26,27 AT 37:A1 = 26:B1 = 37: RETURN
2300  VLIN A1,A1 + 1 AT B1: COLOR= 0: VLIN 30,31 AT 37:A1 = 30:B1 = 37: RETURN
2400  VLIN A1,A1 + 1 AT B1: COLOR= 0: VLIN 34,35 AT 37:A1 = 34:B1 = 37: RETURN
2500  VLIN A1,A1 + 1 AT B1: COLOR= 0: VLIN 34,35 AT 35:A1 = 34:B1 = 35: RETURN
2600  VLIN A1,A1 + 1 AT B1: COLOR= 0: VLIN 34,35 AT 33:A1 = 34:B1 = 33: RETURN
2700  VLIN A1,A1 + 1 AT B1: COLOR= 0: VLIN 30,31 AT 33:A1 = 30:B1 = 33: RETURN
2800  VLIN A1,A1 + 1 AT B1: COLOR= 0: VLIN 26,27 AT 33:A1 = 26:B1 = 33: RETURN
3100 X = 0:Y = 2:XF = 20:YF = 4: RETURN
3200 X =  - 1:Y = 2:XF = 28:YF = 4: RETURN
3300 X =  - 1:Y = 0:XF = 28:YF = 20: RETURN
3400 X =  - 1:Y =  - 2:XF = 28:YF = 36: RETURN
3500 X = 0:Y =  - 2:XF = 20:YF = 36: RETURN
3600 X = 1:Y =  - 2:XF = 12:YF = 36: RETURN
3700 X = 1:Y = 0:XF = 12:YF = 20: RETURN
3800 X = 1:Y = 2:XF = 12:YF = 4: RETURN
4000  VLIN A2,A2 + 1 AT B2: COLOR= 13: VLIN YF,YF + 1 AT XF
4010 A2 = YF:B2 = XF:XF = XF + X:YF = YF + Y: RETURN
5000  COLOR= 3: VLIN Y%,Y% + 9 AT X%: VLIN Y%,Y% + 9 AT X% + 1: VLIN Y%,Y% + 9 AT X% + 2
5005  COLOR= 8
5010  ON Z% GOTO 5100,5200,5300,5400,5500,5600,5700,5800,5900
5050  VLIN Y%,Y% + 9 AT X% + 2: VLIN Y%,Y% + 1 AT X% + 1: VLIN Y%,Y% + 9 AT X%: VLIN Y% + 8,Y% +
 9 AT X% + 1: RETURN
5100  VLIN Y%,Y% + 9 AT X% + 2: VLIN Y% + 2,Y% + 3 AT X% + 1: VLIN Y% + 4,Y% + 5 AT X%: RETURN
5200  VLIN Y%,Y% + 1 AT X%: VLIN Y% + 4,Y% + 9 AT X%: VLIN Y%,Y% + 1 AT X% + 1: VLIN Y% + 4,Y% +
 5 AT X% + 1: VLIN Y% + 8,Y% + 9 AT X% + 1: VLIN Y%,Y% + 5 AT X% + 2: VLIN Y% + 8,Y% + 9 AT X% +
 2: RETURN
5300  VLIN Y%,Y% + 1 AT X%: VLIN Y% + 4,Y% + 5 AT X%: VLIN Y% + 8,Y% + 9 AT X%: VLIN Y%,Y% + 1 A
T X% + 1: VLIN Y% + 4,Y% + 5 AT X% + 1: VLIN Y% + 8,Y% + 9 AT X% + 1: VLIN Y%,Y% + 9 AT X% + 2:
RETURN
5400  VLIN Y%,Y% + 5 AT X%: VLIN Y% + 4,Y% + 5 AT X% + 1: VLIN Y%,Y% + 9 AT X% + 2: RETURN
5500  VLIN Y%,Y% + 1 AT X% + 2: VLIN Y% + 4,Y% + 9 AT X% + 2: VLIN Y%,Y% + 1 AT X% + 1: VLIN Y%
+ 4,Y% + 5 AT X% + 1: VLIN Y% + 8,Y% + 9 AT X% + 1: VLIN Y%,Y% + 5 AT X%: VLIN Y% + 8,Y% + 9 AT
X%: RETURN
5600  VLIN Y%,Y% + 1 AT X% + 2: VLIN Y% + 4,Y% + 9 AT X% + 2: VLIN Y%,Y% + 1 AT X% + 1: VLIN Y%
+ 4,Y% + 5 AT X% + 1: VLIN Y%,Y% + 9 AT X%: VLIN Y% + 8,Y% + 9 AT X% + 1: RETURN
5700  VLIN Y%,Y% + 1 AT X%: VLIN Y%,Y% + 1 AT X% + 1: VLIN Y%,Y% + 9 AT X% + 2: RETURN
5800  VLIN Y%,Y% + 9 AT X% + 2: VLIN Y%,Y% + 1 AT X% + 1: VLIN Y% + 4,Y% + 5 AT X% + 1: VLIN Y%,
Y% + 9 AT X%: VLIN Y% + 8,Y% + 9 AT X% + 1: RETURN
5900  VLIN Y%,Y% + 5 AT X%: VLIN Y% + 8,Y% + 9 AT X%: VLIN Y%,Y% + 1 AT X% + 1: VLIN Y% + 4,Y% +
 5 AT X% + 1: VLIN Y% + 8,Y% + 9 AT X% + 1: VLIN Y%,Y% + 9 AT X% + 2: RETURN
```

```
ÜLIST

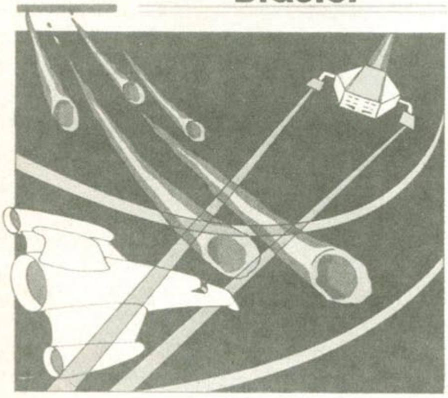
1  REM  **************************************************************
2  REM  *                                                            *
3  REM  *                   SUCHRAETSEL-ERZEUGER                     *
4  REM  *                                                            *
5  REM  *                        Homecomputer                        *
6  REM  *                                                            *
7  REM  **************************************************************
8  REM
10  HIMEM: 16000
20  HOME
30 L$ = "                         "
40 AF = 16000
50  INPUT "FELDBREITE,LAENGE ? ";B,L
60  IF AF + B * L > 38399 THEN  PRINT : PRINT "FELD ZU GROSS ": PRINT : GOTO 50
70  PRINT : PRINT : INPUT "WIEVIEL WOERTER ? ";WW
80  DIM W$(WW + 6)
90  FOR I = AF TO AF + B * L
100  POKE I,0
110  NEXT
120  FOR I = 1 TO WW
130  PRINT : PRINT I;: INPUT ". WORT ? ";W$(I)
140  IF  LEN (W$(I)) > 24 THEN  PRINT : PRINT "WORT ZU LANG ": CALL 64477: GOTO 130
150  IF W$(I) = "C" THEN  PRINT : INPUT "NUMMER,WORT ? ";N,W$(N): GOTO 130
160  NEXT
170 F = 0: FOR I = 2 TO WW: IF  LEN (W$(I)) <  =  LEN (W$(I - 1)) THEN 190
180 W$ = W$(I):W$(I) = W$(I - 1):W$(I - 1) = W$:F = 1
190  NEXT
200  IF F = 1 THEN 170
210  FOR I = 1 TO WW
220  PRINT I,W$(I)
230 LW =  LEN (W$(I)) + 1
240 R =  RND (1) * 8:R =  INT (R) + 1
250  ON R GOSUB 1000,2000,3000,4000,5000,6000,7000,8000
260  NEXT
270  FOR I = 0 TO L - 1: FOR J = 0 TO B - 1
280  IF  PEEK (AF + I * B + J) = 0 THEN  POKE AF + I * B + J,46
290  NEXT J,I
300  PR# 1
320  FOR I = 0 TO L - 1: FOR J = 0 TO B - 1
330  PRINT  CHR$ ( PEEK (AF + I * B + J));" ";
340  NEXT : PRINT : NEXT
350  PRINT : PRINT
360  FOR I = 1 TO WW STEP 5
370  FOR J = 0 TO 3
380  PRINT W$(I + J); LEFT$ (L$,25 -  LEN (W$(I + J)));
390  NEXT : PRINT : NEXT
400  FOR I = 0 TO L - 1: FOR J = 0 TO B - 1
410  IF  PEEK (AF + I * B + J) = 46 THEN  POKE AF + I * B + J, RND (1) * 26 + 65
420  NEXT J,I
430  PRINT  CHR$ (12)
440  FOR I = 0 TO L - 1: FOR J = 0 TO B - 1
450  PRINT  CHR$ ( PEEK (AF + I * B + J));" ";
460  NEXT : PRINT : NEXT
470  PRINT : PRINT
480  FOR I = 1 TO WW STEP 5
490  FOR J = 0 TO 3
500  PRINT W$(I + J); LEFT$ (L$,25 -  LEN (W$(I + J)));
510  NEXT : PRINT : NEXT
520  PR# 0
999  END
1000  REM  ** RICHTUNG NO **
1010 X =  RND (1) * (B - 1 - LW)
1020 Y =  RND (1) * (L - 1 - LW) + LW: IF X < 0 OR Y < 0 THEN  POP : GOTO 240
1025  IF X > B - 1 OR Y > L - 1 THEN  POP : GOTO 240
1030 X =  INT (X):Y =  INT (Y):X1 = X:Y1 = Y
1040  FOR J = 1 TO LW - 1
1050  IF  PEEK (AF + Y * B + X) >  < 0 THEN  IF  PEEK (AF + Y * B + X) >  <  ASC ( MID$ (W$(I),J
,1)) THEN  POP : GOTO 240
1060 X = X + 1:Y = Y - 1
1070  NEXT
1080 X = X1:Y = Y1
1090  FOR J = 1 TO LW - 1
1100  POKE AF + Y * B + X, ASC ( MID$ (W$(I),J,1))
1110 X = X + 1:Y = Y - 1
1120  NEXT
1130  RETURN
2000  REM  ** RICHTUNG O **
2010 X =  RND (1) * (B - 1 - LW)
2020 Y =  RND (1) * (L - 1): IF X < 0 OR Y < 0 THEN  POP : GOTO 240
2025  IF X > B - 1 OR Y > L - 1 THEN  POP : GOTO 240
2030 X =  INT (X):Y =  INT (Y):X1 = X:Y1 = Y
2040  FOR J = 1 TO LW - 1
2050  IF  PEEK (AF + Y * B + X) >  < 0 THEN  IF  PEEK (AF + Y * B + X) >  <  ASC ( MID$ (W$(I),J
,1)) THEN  POP : GOTO 240
2060 X = X + 1
2070  NEXT
2080 X = X1
2090  FOR J = 1 TO LW - 1
2100  POKE AF + Y * B + X, ASC ( MID$ (W$(I),J,1))
2110 X = X + 1
2120  NEXT
2130  RETURN
3000  REM  ** RICHTUNG SO **
3010 X =  RND (1) * (B - 1 - LW)
3020 Y =  RND (1) * (L - 1 - LW): IF X < 0 OR Y < 0 THEN  POP : GOTO 240
3025  IF X > B - 1 OR Y > L - 1 THEN  POP : GOTO 240
3030 X =  INT (X):Y =  INT (Y):X1 = X:Y1 = Y
3040  FOR J = 1 TO LW - 1
3050  IF  PEEK (AF + Y * B + X) >  < 0 THEN  IF  PEEK (AF + Y * B + X) >  <  ASC ( MID$ (W$(I),J
,1)) THEN  POP : GOTO 240
3060 X = X + 1:Y = Y + 1
3070  NEXT
3080 X = X1:Y = Y1
3090  FOR J = 1 TO LW - 1
3100  POKE AF + Y * B + X, ASC ( MID$ (W$(I),J,1))
3110 X = X + 1:Y = Y + 1
3120  NEXT
3130  RETURN
4000  REM  ** RICHTUNG S **
4010 X =  RND (1) * (B - 1)
4020 Y =  RND (1) * (L - 1 - LW): IF X < 0 OR Y < 0 THEN  POP : GOTO 240
4025  IF X > B - 1 OR Y > L - 1 THEN  POP : GOTO 240
4030 X =  INT (X):Y =  INT (Y):X1 = X:Y1 = Y
4040  FOR J = 1 TO LW - 1
4050  IF  PEEK (AF + Y * B + X) >  < 0 THEN  IF  PEEK (AF + Y * B + X) >  <  ASC ( MID$ (W$(I),J
,1)) THEN  POP : GOTO 240
4060 Y = Y + 1
4070  NEXT
4080 Y = Y1
4090  FOR J = 1 TO LW - 1
4100  POKE AF + Y * B + X, ASC ( MID$ (W$(I),J,1))
4110 Y = Y + 1
4120  NEXT
4130  RETURN
5000  REM  ** RICHTUNG SW **
5010 X =  RND (1) * (B - 1 - LW) + LW
5020 Y =  RND (1) * (L - 1 - LW): IF X < 0 OR Y < 0 THEN  POP : GOTO 240
5025  IF X > B - 1 OR Y > L - 1 THEN  POP : GOTO 240
5030 X =  INT (X):Y =  INT (Y):X1 = X:Y1 = Y
```

```
5040  FOR J = 1 TO LW - 1
5050  IF  PEEK (AF + Y * B + X) >  < 0 THEN  IF  PEEK (AF + Y * B + X) >  <  ASC ( MID$ (W$(I),J
,1)) THEN  POP : GOTO 240
5060 X = X - 1:Y = Y + 1
5070  NEXT
5080 Y = Y1:X = X1
5090  FOR J = 1 TO LW - 1
5100  POKE AF + Y * B + X, ASC ( MID$ (W$(I),J,1))
5110 X = X - 1:Y = Y + 1
5120  NEXT
5130  RETURN
6000  REM   ** RICHTUNG W **
6010 X =  RND (1) * (B - 1 - LW) + LW
6020 Y =  RND (1) * (L - 1): IF X < 0 OR Y < 0 THEN  POP : GOTO 240
6025  IF X > B - 1 OR Y > L - 1 THEN  POP : GOTO 240
6030 X =  INT (X):Y =  INT (Y):X1 = X:Y1 = Y
6040  FOR J = 1 TO LW - 1
6050  IF  PEEK (AF + Y * B + X) >  < 0 THEN  IF  PEEK (AF + Y * B + X) >  <  ASC ( MID$ (W$(I),J
,1)) THEN  POP : GOTO 240
6060 X = X - 1
6070  NEXT
6080 X = X1
6090  FOR J = 1 TO LW - 1
6100  POKE AF + Y * B + X, ASC ( MID$ (W$(I),J,1))
6110 X = X - 1
6120  NEXT
6130  RETURN
7000  REM   ** RICHTUNG NW **
7010 X =  RND (1) * (B - 1 - LW) + LW
7020 Y =  RND (1) * (L - 1 - LW) + LW: IF X < 0 OR Y < 0 THEN  POP : GOTO 240
7025  IF X > B - 1 OR Y > L - 1 THEN  POP : GOTO 240
7030 X =  INT (X):Y =  INT (Y):X1 = X:Y1 = Y
7040  FOR J = 1 TO LW - 1
7050  IF  PEEK (AF + Y * B + X) >  < 0 THEN  IF  PEEK (AF + Y * B + X) >  <  ASC ( MID$ (W$(I),J
,1)) THEN  POP : GOTO 240
7060 X = X - 1:Y = Y - 1
7070  NEXT
7080 X = X1:Y = Y1
7090  FOR J = 1 TO LW - 1
7100  POKE AF + Y * B + X, ASC ( MID$ (W$(I),J,1))
7110 X = X - 1:Y = Y - 1
7120  NEXT
7130  RETURN
8000  REM   ** RICHTUNG N **
8010 X =  RND (1) * (B - 1)
8020 Y =  RND (1) * (L - 1 - LW) + LW: IF X < 0 OR Y < 0 THEN  POP : GOTO 240
8025  IF X > B - 1 OR Y > L - 1 THEN  POP : GOTO 240
8030 X =  INT (X):Y =  INT (Y):X1 = X:Y1 = Y
8040  FOR J = 1 TO LW - 1
8050  IF  PEEK (AF + Y * B + X) >  < 0 THEN  IF  PEEK (AF + Y * B + X) >  <  ASC ( MID$ (W$(I),J
,1)) THEN  POP : GOTO 240
8060 Y = Y - 1
8070  NEXT
8080 Y = Y1
8090  FOR J = 1 TO LW - 1
8100  POKE AF + Y * B + X, ASC ( MID$ (W$(I),J,1))
8110 Y = Y - 1
8120  NEXT
8130  RETURN
```

Great Games From Piccadilly Software Inc.

Piccadilly Software Presents

Star Blaster



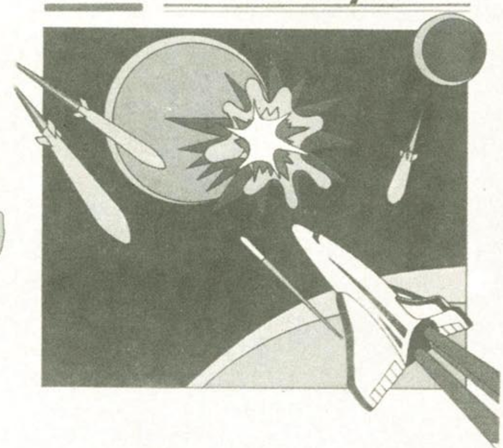
Piccadilly Software Presents

Falcons



Piccadilly Software Presents

Warp Destroyer



Piccadilly Software Presents

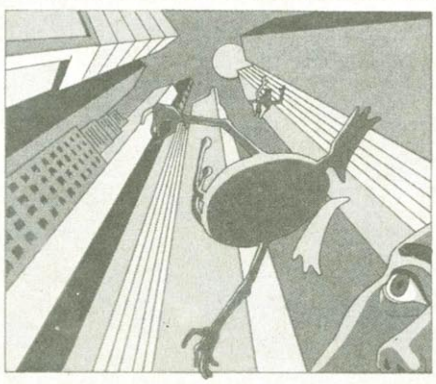
Survival



COMING IN THE 2ND HALF OF 1982

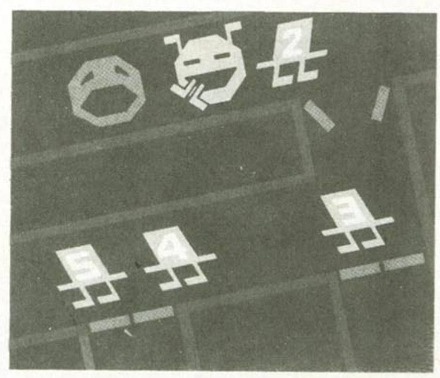
Piccadilly Software Presents

Suicide!



Piccadilly Software Presents

Succession



Available at your local dealer.

Requires: APPLE II/APPLE II PLUS with 48K in Dos 3.3 or 3.2 • APPLE III

Piccadilly Software Inc. • 89 Summit Avenue • Summit, N.J. 07901 • (201) 277-1020

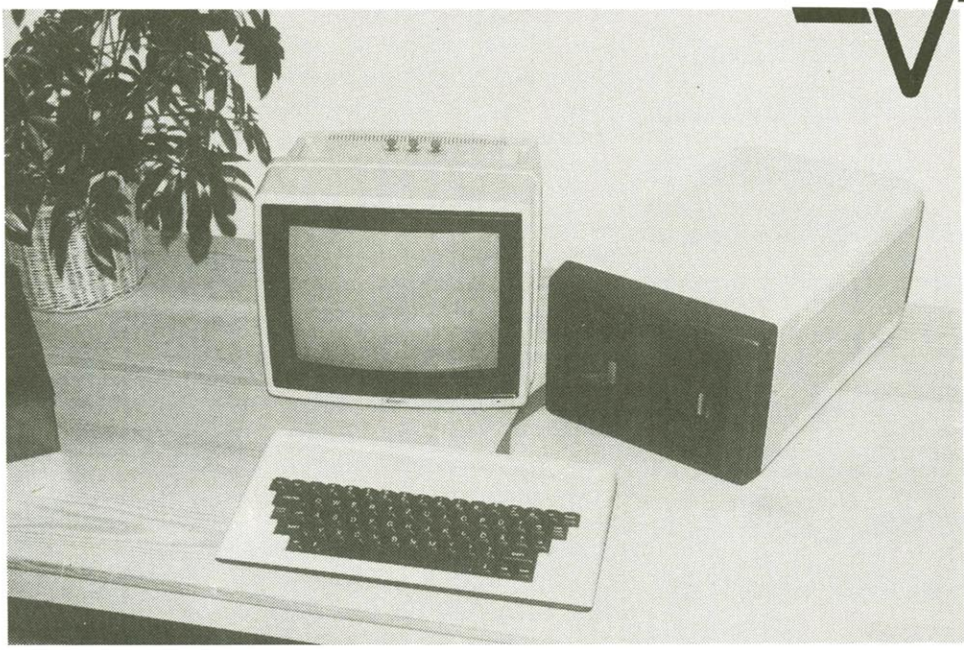
If you're unable to obtain Piccadilly Games from your local dealer, send $29.95 per game plus $1.50 per order for postage and handling. NJ residents add 5% sales tax. Outside US, send $3.00 per order for shipping.

*Apple is a registered trademark of Apple Computer Inc.

ORANGE 2 COMPUTER (voll APPLE kompatibel)

Groß-/Kleinschreibung, Umlaute bei allen Ausführungen

ARHELGER electronic



Komplettsystem wie abgebildet	ab DM **4499,–**
Bausatz: Hauptplatine 48K-RAM und alle Teile, fertiges Netzteil, Cherry-Tastatur (4 Cursortasten, 2 Editiertasten, Hometaste, Gehäuse)	DM **1388,–**
Fertiggerät: wie oben, jedoch fertig aufgebaut, **Orange 2** 6 Monate Garantie	nur DM **1688,–**

Gehäuse für Rechner und 2 Drives auf Anfrage.

Profitastatur zur Textverarbeitung:
Zehnerblock, Funktionstasten, Cursortasten, Umlaute, anschlußfertig im Gehäuse. Aufpreis **nur DM 488,–**

PERIPHERIE:

NEC-8023 BC m. Graphikinterf.	DM **1950,–**
BMC-Monitor Typ 12 grün geätzte Röhre	DM **540,–**
Zenit-Monitor ZUM 121 (LOW COST)	DM **305,–**
16K Language Card	DM **220,–**
80 Zeichenkarte m. 40/80 Z. Softswitch	DM **495,–**
CP/M Z-80 CPU-Karte	DM **375,–**
Verbatim Datalife Disketten 5¼" (10)	DM **79,–**

Diskettenlaufwerke und weiteres Zubehör auf Anfrage!

Alle Preise verstehen sich inkl. 13 % MwSt.
Weitere Informationen gegen DM 1,50 in BM.

Versand erfolgt unfrei.

3544 WALDECK-SACHSENHAUSEN WINTERHAGEN 2
TELEFON 05634-1724 TELEX 991160 AEW D

DISKETTENSCHUTZ FÜR APPLE II

Wohl jeder Apple Benutzer, der mit Floppy-Laufwerken arbeitet, kennt sogenannte "geschützte" Disketten. Die Daten auf diesen Disketten sind in einem anderen, nicht standardisiertem Code abgelegt, der vom normalen DOS 3.3 nicht gelesen werden kann. Das Resultat eines CATALOG-Versuches ist dann ein I/O-ERROR. Wie man solche geschützten Disketten selber herstellen kann, wird in diesem Artikel beschrieben.

Die Grundlagen für dieses Programm entstammen den schon in Heft 3/83 erwähnten Buch 'Beneath Apple Dos', das jedem, der sich mit den Hintergründen des DOS 3.2 und 3.3 befassen will, empfohlen werden kann. Dort kann man u. a. auch lesen, daß ein Sektor auf Diskette folgendes Format hat:

! Adreß-Feld ! Zwischenraum ! Datenfeld ! Zwischenraum !

Vor dem ersten Adreßfeld einer Spur steht noch ein weiterer, längerer Zwischenraum.

Was bedeuten diese Felder? Nun, beim Diskettenzugriff muß der Computer erkennen, welcher Sektor der sich drehenden Diskette gerade am Kopf vorbeiläuft. Dazu dient das Adreßfeld, das folgenden Aufbau hat:

! D5 AA 96 ! Volume Nr. ! Spur ! Sektor ! Prüfsumme ! DE AA EB !

Dieses Feld wird einmal beim Initialisieren auf die Diskette geschrieben, und bleibt dann bis zum nächsten Formatieren erhalten. Die Zwischenräume bestehen aus $FFs, die allerdings um zwei Bits verlängert sind, sodaß sich ein 10-Bit Code ergibt. Anhand dieser Bitfolge auf Diskette kann das Laufwerk synchronisiert werden.

Das Datenfeld ist der eigentliche, interessante Teil. Es hat folgenden Aufbau:

! D5 AA AD ! 342 Byte Daten ! Prüfsumme ! DE AA EB !

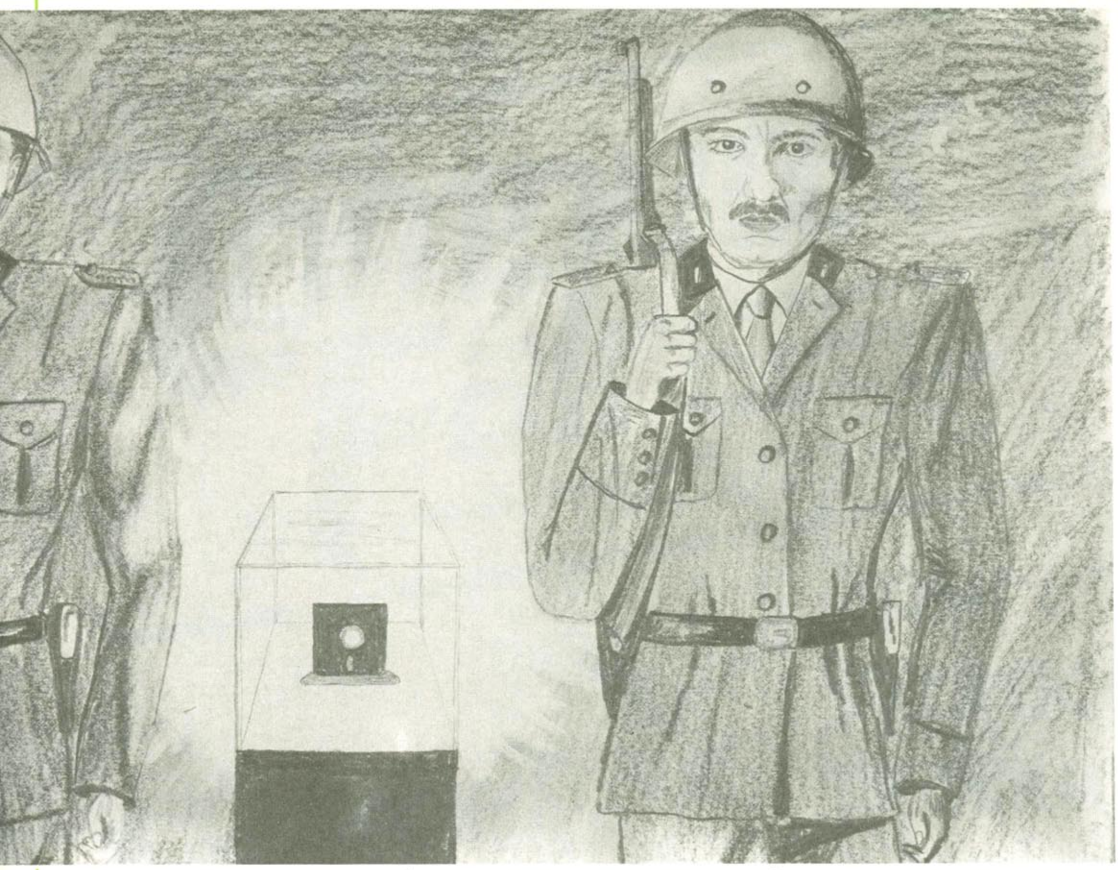
Anstatt der erwarteten 256 Bytes werden 342 Bytes aufgezeichnet. Das liegt daran, daß die Daten nicht direkt als 8-Bit-Code geschrieben werden können, sondern es sind nur 6 Bits relevant. Die 256 Byte mal 8 Bit müssen also in 342 Byte mal 6 Bits aufgeteilt werden. Wie das gemacht wird, steht in oben erwähnten Werk.

Das Datenfeld, bestehend aus 350 Byte (342 Byte Daten + 2 Byte Prüfsumme + 3 Byte Vorspann und Nachspann) wird nun bei jedem Schreibzugriff neu auf Diskette aufgezeichnet. Wenn es jetzt gelänge, den Vor- oder den Nachspann zu ändern, würde die RWTS (Read or Write a Track and Sector, Sektoren Lese-/Schreib-) Routine das Datenfeld nicht mehr erkennen, und den bekannten I/O-ERROR melden.

Und genau das macht das vorliegende Programm. Jeder einzelne Sektor der Diskette wird ganz normal eingelesen und dann, nachdem zwei Bytes in der RWTS geändert wurden, wieder auf Diskette zurückgeschrieben. Diese Änderung in zwei Bytes bewirkt aber, daß beim Zurückschreiben auf Diskette nicht mehr D5 AA AD als Vorspann des Datenfeldes aufgezeichnet wird, sondern D5 AA CE. Damit ist die o. g. Bedingung erreicht, und die Diskette ist nicht mehr lesbar.



Wenn man das nun mit allen Sektoren der Diskette macht, wird sie nicht nur vom normalen DOS aus unlesbar, sie bootet auch nicht mehr, kann sich also selber auch nicht lesen. Beim Bootvorgang sucht der Controller nämlich als Beginn des Datenfeldes ebenfalls D5 AA AD, findet diese Bitfolge nicht, und das Laufwerk hört nicht auf zu laufen. Also müssen die ersten 10 Sektoren der Diskette, auf denen die RWTS steht, im normalen Format aufgezeichnet werden. Wird auch das erledigt, so bootet die Diskette immer noch nicht. Beim Booten wird der Controller jetzt den ersten Sektor der Diskette laden, dieser lädt dann neun weitere Sektoren, die die RWTS enthalten. Diese ersten 10 Sektoren sind also noch im Standard-Format aufgezeichnet. Nun besteht die Aufgabe der eben geladenen



RWTS darin, 25 weitere Sektoren zu laden, die das eigentliche DOS enthalten, und im geschützten Format auf Diskette stehen. Da diese Daten durch das Verändern von zwei Bytes geschrieben wurden, müssen diese zwei Byte jetzt auch in der eben geladenen RWTS geändert werden. Sinnvoller ist es jedoch, die Bytes nicht nach dem Laden, sondern schon vorher, auf Diskette, zu ändern. Wenn dann auch diese Modifikation durchgeführt wurde, läuft die Diskette einwandfrei, und verhält sich wie beim normalen Booten. Auf Standard-Disketten kann allerdings nicht zugegriffen werden, da diese RWTS ja nur das geänderte Format akzeptiert. Außerdem können Disketten zwar initialisiert werden, sind aber nicht bootfähig, da die ersten 10 Sektoren auch im modifizierten Format geschrieben werden. Mit vorliegendem Programm können allerdings Disketten im geschützten Format initialisiert werden, die auch booten.

Nach der Theorie folgt nun die Praxis: Das Programm selbst wurde in Basic geschrieben, enthält aber relativ viel Maschinensprache. Die Maschinenprogramme selbst sind in DATA-Zeilen abgelegt, so daß das Eintippen von Maschinenprogrammen im Monitor entfällt. Nach dem Starten des Programms hat der Benutzer die Auswahl, ob er eine leere Diskette im geschützten Format initialisieren will (dabei bleibt wie oben geschildert die Bootmöglichkeit erhalten), ob er auf eine bestehende Diskette in das geschützte Format umwandeln will, oder ob auf eine geschützte Diskette wieder das Standard-DOS geschrieben werden soll. Nach Beantwortung dieser Frage muß eingegeben werden, auf welchen Laufwerk (1 oder 2) der Vorgang stattfinden soll. Nun wird geprüft ob und wenn ja, welches DOS auf der Diskette vorhanden ist. Sollte auf der Diskette ein unbekanntes DOS aufgezeichnet sein, so wird (außer beim Neuinitialisieren) das Programm abgebrochen. Ebenfalls wird erkannt, ob das gewünschte DOS vielleicht schon auf der Diskette vorhanden ist, und gegebenenfalls das Programm beendet. Als besonderes 'Feature' ergibt sich die Möglichkeit, zwischen dem normalen und dem geschützten DOS hin- und herzuschalten, so daß z. B. Dateien kopiert werden können.

Das Maschinenprogramm, das in der DATA-Zeile 230 steht, hat die Aufgabe, die RWTS oder die BLKRWTS (Lesen und Schreiben ganzer Blöcke, wird z. B. in den Zeilen 20080-20120 angewandt), aufzurufen. Das Programm, das in den DATA-Zeilen 460-690 steht, dient zum Umschalten zwischen den beiden DOS-Versionen.

Durch Veränderung des Wertes von CH in Zeile 100 können auch andere Bytes als CE für das geänderte AD genommen werden. Allerdings müssen diese Bytes bestimmte Bedingungen erfüllen. Insgesamt sind 66 Bytes zulässig, die man in 'Beneath Apple Dos', Seite 3-21 nachlesen kann.

Es muß noch gesagt werden, daß dieses Programm zwar ein Kopieren mit COPYA oder FID unmöglich macht, jedoch den Zugriff zur Diskette nicht verwehrt. So kann zum Beispiel direkt nach dem Booten der geschützten Diskette Control-C gedrückt werden, was einen Sprung ins Basic bewirkt. Wie man das verhindern kann, steht im nächsten Heft. Dort werden dann auch ausführlich alle Möglichkeiten diskutiert, einen RESET abzufangen.

Kurze Beschreibung des Aufbaus des Programms:

Zeile(n) Bedeutung

10 Erniedrigen von HIMEM, Merken des alten Wertes

20-150 Menü und Definition der Variablen

160-190 Programmabbruch, wenn gewünschte DOS-Version = vorhandene DOS-Version

200-220 Diskette formatieren

230-360 Sektoren 0-9 wieder in normalem Format schreiben

370-690 Maschinenprogramm zum Umschalten laden + initialisieren

20000-20260 Ändern einer bestehenden Diskette:

20000-20040 Variablen initialisieren

20050-20150 jeweils 7 Spuren laden und im anderen Format schreiben (Ausnahme: Zurückschreiben des normalen Dos, Sektoren 0-9)

20160-20230 Ändern der RWTS, Sektor 2, Byte 93 + 252

30000-30050 Feststellen der gespeicherten DOS-Version

Jörg Bliesener

```
]LIST

10 HM =  PEEK (115) + 256 *  PEEK (116): HIMEM: 8192: TEXT
20  HOME : INVERSE : PRINT "                                  S P E C I A L   D O S
       C R E A T E R  -------------------------------------
                        BY JOERG BLIESENER              ";
30  PRINT "                                        ";
40  NORMAL : VTAB 12: PRINT "WHICH MODE": PRINT : PRINT "U(PDATE DISK TO SPECIAL DOS": PRINT
     "C(REATE NEW DISK (DESTROYS DATA!)": PRINT "R(EWRITE NORMAL DOS": PRINT : PRINT "YOUR
     CHOICE: ->";
50  GET M$: IF M$ <  > "U" AND M$ <  > "C" AND M$ <  > "R" THEN 50
60 DN$ = "SPECIAL"
70 WR = 47197
80 RD = 47356
90 D$ =  CHR$ (4)
100 CH = 206:NO = 173:OG = NO:OS = CH
110  IF M$ = "R" THEN H = CH:CH = NO:NO = H:DN$ = "NORMAL "
120  HOME : VTAB 12
130  PRINT : PRINT "WHICH DRIVE (1/2) ?";
140  GET DR$: IF DR$ <  > "1" AND DR$ <  > "2" THEN 140
150  HOME : VTAB 12: PRINT "INSERT DISK TO CONTAIN "DN$" DOS IN   DRIVE "DR$" AND PRESS <R
     ETURN>": POKE  - 16368,0: WAIT  - 16384,128: GET A$: POKE  - 16368,0
160 DR =  VAL (DR$): GOSUB 30000
170  IF DS$ = "NORMAL" AND M$ = "R" THEN  HOME : VTAB 12: PRINT "DISK CONTAINS ALREADY NOR
     MAL DOS!": POKE 35,24: END
180  IF DS$ = "SPECIAL" AND (M$ = "C" OR M$ = "U") THEN  HOME : VTAB 12: PRINT "DISK CONTA
     INS ALREADY SPECIAL DOS!": POKE 35,24: END
185  IF DS$ = "UNKNOWN" AND (M$ = "U" OR M$ = "R") THEN  HOME : VTAB 12: PRINT "DISK CONTA
     INS UNKNOWN DOS. UPDATING IM- POSSIBLE !": POKE 35,24: END
190 DR =  VAL (DR$): IF M$ = "U" OR M$ = "R" THEN 230
200  HOME : VTAB 12: PRINT "NOW INITIALIZING DISK. PLEASE WAIT FOR  APPROX. 40 SECONDS"
210  POKE WR,CH: POKE RD,CH: PRINT D$"INIT HELLO,D"DR$
220  PRINT D$"DELETE HELLO,D"DR$
230  DATA  32,227,3,133,7,132,6,96,165,7,164,6,32,217,3,96
240  FOR I = 768 TO 783: READ J: POKE I,J: NEXT
250 T = 0: CALL 768
260 IOB =  PEEK (6) + 256 *  PEEK (7)
270  IF M$ = "U" OR M$ = "R" THEN 20000
280  FOR S = 0 TO 9
290  POKE IOB + 2,DR: POKE IOB + 3,0: POKE IOB + 4,T: POKE IOB + 5,S: POKE IOB + 8,0: POKE
     IOB + 9,2: POKE IOB + 12,1: POKE IOB + 13,0
300  POKE WR,NO: POKE RD,CH
310  CALL 776
320  POKE IOB + 2,DR: POKE IOB + 3,0: POKE IOB + 4,T: POKE IOB + 5,S: POKE IOB + 8,0: POKE
     IOB + 9,2: POKE IOB + 12,2: POKE IOB + 13,0
330  CALL 776
340  IF M$ = "I" OR M$ = "F" THEN  PRINT ".";
350  NEXT
360  GOSUB 30000: FOR I = 1 TO 16: READ D: NEXT
370  POKE 971,CH: POKE 974,NO: POKE 975,255
380  FOR I = 768 TO 894: READ P: POKE I,P: NEXT
390  CALL 768
400  HOME : VTAB 9
410  PRINT "YOU CAN NOW FLIP BETWEEN THE NORMAL AND THE SPECIAL DOS WITH THE &-COMMAND TO
       TRANSFER FILES TO YOUR NEW DISK."
420  PRINT : PRINT "THE MODE WILL BE DISPLAYED": PRINT : PRINT "]&"
430  POKE WR,NO: POKE RD,NO
440  PRINT "NORMAL DOS"
450  POKE 35,24: HIMEM: HM: END
460  DATA  169,76,141,245,3
470  DATA  169,16,141,246,3
480  DATA  169,3,141,247,3,96
490  DATA  173,207,3,208,28
500  DATA  173,206,3,141,93
510  DATA  184,141,252,184
520  DATA  169,255,141,207,3
530  DATA  32,83,3,206,207
540  DATA  210,205,193,204
550  DATA  160,0,76,75,3,173
560  DATA  203,3,141,93,184
570  DATA  141,252,184,169,0
580  DATA  141,207,3,32,83,3
590  DATA  211,208,197,195
600  DATA  201,193,204,160,0
610  DATA  32,83,3,196,207
620  DATA  211,0,96,104,133,6
630  DATA  104,133,7,230,6
640  DATA  240,1,44,230,7,160
650  DATA  0,177,6,240,15,9
660  DATA  128,32,237,253,230
670  DATA  6,240,1,44,230,7
680  DATA  76,98,3,230,6,240
690  DATA  1,44,230,7,108,6,0
20000  REM  **** UPDATE EXISTING DISK ****
20010  IF M$ = "R" THEN  HOME : VTAB 12: PRINT "REWRITING DOS.";: GOTO 20030
20020  HOME : VTAB 12: PRINT "UPDATING DISK.";
20030  POKE RD,NO: POKE WR,NO
20040  POKE 781,147: POKE 782, PEEK (7)
20050  FOR I = 34 TO 0 STEP  - 7
20060  POKE RD,NO: POKE WR,NO
20070  IF I = 6 AND M$ = "R" THEN AN = 102: GOTO 20090
20080 AN = 112
20090  POKE IOB + 2,DR: POKE IOB + 3,0: POKE IOB + 4,I: POKE IOB + 5,15: POKE IOB + 8,0: POKE
      IOB + 9,144: POKE IOB + 12,1: POKE IOB + 13,0: POKE IOB - 7,AN
20100  CALL 776
20110  POKE RD,CH: POKE WR,CH
20120  POKE IOB + 2,DR: POKE IOB + 3,0: POKE IOB + 4,I: POKE IOB + 5,15: POKE IOB + 8,0: POKE
      IOB + 9,144: POKE IOB + 12,2: POKE IOB + 13,0: POKE IOB - 7,AN
20130  PRINT ".";
20140  CALL 776
20150  NEXT
20160  IF M$ = "R" THEN  POKE RD,CH: POKE WR,CH
20170  POKE 781,217: POKE 782,3: POKE IOB + 2,DR: POKE IOB + 3,0: POKE IOB + 4,0: POKE IOB
       + 5,2: POKE IOB + 8,0: POKE IOB + 9,32: POKE IOB + 12,1: POKE IOB + 13,0
20180  CALL 776
20190 X = 8192
20200  POKE X + 93,CH: POKE X + 252,CH
20210  POKE 781,217: POKE 782,3: POKE IOB + 2,DR: POKE IOB + 3,0: POKE IOB + 4,0: POKE IOB
       + 5,2: POKE IOB + 8,0: POKE IOB + 9,32: POKE IOB + 12,2: POKE IOB + 13,0
20220  CALL 776
20230  PRINT ".";
20240  IF M$ = "U" THEN M$ = "I": RESTORE : GOTO 230
20250  GOSUB 30000
20260  HOME : VTAB 12: PRINT "NORMAL DOS REWRITTEN": POKE 35,24: HIMEM: HM: END
30000  RESTORE : FOR I = 768 TO 783: READ J: POKE I,J: NEXT : CALL 768
30010 IOB =  PEEK (6) + 256 *  PEEK (7): POKE RD,OG: POKE WR,OG: POKE IOB + 2,DR: POKE IOB
       + 3,0: POKE IOB + 4,0: POKE IOB + 5,2: POKE IOB + 8,0: POKE IOB + 9,2: POKE IOB + 12
      ,1: POKE IOB + 13,0: CALL 776
30020  IF  PEEK (605) = OG THEN DS$ = "NORMAL"
30030  IF  PEEK (605) = OS THEN DS$ = "SPECIAL"
30035  IF  PEEK (605) <  > OG AND  PEEK (605) <  > OS THEN DS$ = "UNKNOWN"
30040  HTAB 1: RESTORE : POKE 35,24
30050  VTAB 21: PRINT "DISK IN DRIVE "DR$" CONTAINS": FLASH : PRINT DS$;: NORMAL : PRINT "
       DOS.": POKE 35,20: RETURN
```

HOW TO GROW UP TO A REAL COMPUTER



If you're enthusiastic about microcomputing, sooner or later you'll ask yourself the question . . . "where do I go from here?"

This is particularly true if you own a micro with limited expansion and hardware peripheral options . . . like the ZX81.

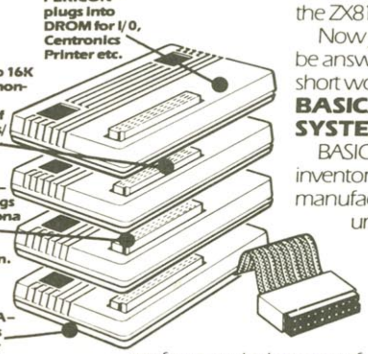
Now your question can be answered in three short words:

BASICARE MICRO SYSTEMS

BASICARE are the inventors, developers and manufacturers of a totally unique microcomputing upgrade system.

It's the sort of system others have dreamed about . . . a series of separate modules that perform a whole range of microcomputing functions that simply (and firmly) stack together.

How BASICARE build into a complete computing package.



If you're confused . . . don't be . . . you only have to think of it in terms of the way Hi-FI equipment has evolved.

You buy the hardware you want and add to the system! Each module may have a separate function or integrate functions. And when you want more . . . you add more!

In short you can develop a whole range of hardware options that fit together to form a complete package . . . "Computing" in the real sense of the word!

And what a package!

Apart from it's good looks and stability under working conditions, BASICARE MICRO SYSTEMS offers a fantastic range of micro options for ZX81 users.

Of course, such a system needs a starting point from which to expand. The heart of BASICARE'S system is a unique computer interface which we call PERSONA.

This one unit simply plugs into your ZX81 without modification and acts as the "brains" of the whole operation.

Thereafter you choose how you want to expand your micro by simply plugging-in more modules.

Look at the choice you have:

PERSONA — An interface module to enable an ORGANIC MICRO to grow on the ZX81.
MINIMAP — A memory mapping device to extend the address space of the ZX81 from 64K bytes to 1 M bytes.
RAM 08 — A low cost, low power memory expandable from 2K to 8K.
RAM 16 — 16K Add on memory at remarkably low cost.
RAM 64 — A TRUE 64K Add on memory.
DROM — Ultra low power memory backed by rechargeable battery for non-volatile storage of programs and data.
TOOLKIT — A module fully socketed to take up to 8K bytes of utilities in EPROM/ROM.
PERICON a — A general purpose, user programmable device providing 24 lines of input/output.
PERICON b — 24 lines of heavy duty output to access and control the outside world.
PERICON c — A module to drive 80 column printer with Centronics type parallel interface.
USERFONT option — User definable characters available for RAM 08, DROM and TOOLKIT.

Of course, there's lots more BASICARE Modules under development including joysticks, EPROM Programmer, Floppy Disc Controller and much more.

BASICARE is the sort of system that GROWS when you grow . . . and remember, when you change your micro, you will be able to change to a PERSONA unit to suit your new equipment!

In short, BASICARE will serve you forever . . . no matter how big you want to grow!

"As space is limited to describe BASICARE products, we have produced a fully documented brochure . . . send for it today (enclosing stamp please).



BASICARE MICRO SYSTEMS are available by mail simply by sending today together with cheque/ P.O./Access or Barclaycard No. Please indicate clearly your exact requirements.

Name ________________

Address ________________

PERSONA	@ **£30.25**	DROM(2K)	@ **£39.50**
MINIMAP	@ **£35.95**	TOOLKIT	@ **£22.20**
RAM8 (2K)	@ **£24.50**	PERICON A	@ **£27.90**
RAM16	@ **£26.75**	PERICON B	@ **£33.75**
RAM64	@ **£76.25**	PERICON C	@ **£41.75**

Options: USERFONT @ £8.00,
Add. 2K for RAM8 @ **£6.50**, Add. 2K for DROM @ **£7.50**

All prices include VAT, postage and packing in the U.K. (Overseas allow at least 15% for surface mail).

Post today to: BASICARE MICROSYSTEMS LTD., 5 Dryden Court, London SE11 4NH or Phone: 01-735 6408.

BASICARE MICRO SYSTEM

Standard-Kommandos für ZX-Programme

Eingangsbemerkungen, die die Inbetriebnahme aller abgedruckten ZX-81-Programme erklären.

GOTO 5 Initialisieren

Mit GOTO 5 wird ein Testlauf gestartet. Wenn keine Tippfehler vorhanden sind, läuft das Programm fehlerfrei ab. Meist wird hier ein Initialisierungsunterprogramm aufgerufen, in dem vorbereitende Rechnungen jeweils nur einmal durchgeführt werden. Das geschieht meist im FAST-Modus und kann einige Zeit (manchmal bis zu einer Minute) in Anspruch nehmen.
Vor dem nächsten Kommando (GOTO 3: SAVE Programm) wird in jedem Falle mit BREAK der Programmablauf gestoppt.
Falls sich durch Tippfehler ein Programmabbruch ergibt oder der Benutzer nach festgestellten Programmfehlern selbständig mit BREAK unterbrach, muß jedesmal GOTO 5 wiederholt werden.

GOTO 3 Programm auf Kassette speichern

Alle Variablen werden vom Programm gelöscht, was die Kassettenlaufzeit verkürzt. Anschließend wird das Programm auf Kassette aufgenommen.

GOTO 1 Neustart

Nach jedem BREAK kann das Programm mit GOTO 1 neu gestartet werden.

Kein RUN

Vermeiden Sie RUN-Kommandos. Nach versehentlichem RUN müssen Sie mit GOTO 5 neu initialisieren!

Basic-Programme, die ein Maschinen-Programm enthalten, beginnen mit einem REM-Statement, hinter dem mindestens so viele Zeichen stehen müssen, wie das Maschinen-Programm Bytes enthält.
In den abgedruckten Programm-Listings ist aus Platzgründen hinter dem REM jeweils nur die Anzahl der Bytes invers angegeben.
Nützlich erweist es sich, die Zahlenfolge 1 bis 0 periodisch zu wiederholen und bei vollen Hundertern die erste Ziffer invers zu tippen. Dies erleichtert das ständige Mitzählen.
Beim Aufbau langer REM-Zeilen können zum Schluß lange Wartezeiten auftreten, die man durch vorheriges Umschalten in den FAST-Modus etwas abkürzen kann.
Um das Maschinen-Programm in den REM-Zeilen unterzubringen verwendet man ein Loader-Programm, das nach seiner Benutzung wieder gelöscht wird. Mit ihm werden die Bytes in Hexadezimalform eingegeben. Der abgedruckte Loader läßt die Eingabe mehrerer Bytes gleichzeitig zu.

```
1000 LET X=16514
1010 INPUT A$
1020 POKE X,16*CODE A$+CODE A$(2
)-476
1030 LET X=X+1
1040 PRINT A$( TO 2);
1050 LET A$=A$(3 TO )
1060 IF A$="" THEN GOTO 1010
1070 GOTO 1020
```

Eine andere Technik zum Einfügen eines Maschinenprogramms speichert alle Bytes zunächst in einem String des Basic-Programms. Eine Ladeschleife - ähnlich dem Loader-Programm - übernimmt das Einfügen in die REM-Zeilen. Die Stringtechnik hat den Vorteil, daß man das Maschinenprogramm in das Basic-Programm integriert in Listen aufbewahren kann. Außerdem erleichtert es die Korrektur eventuell fehlerhaft eingegebener Bytes und macht das Loader-Programm überflüssig.
Nach dem Ladevorgang befindet sich hinter dem REM ein Gewimmel von Zeichen und Befehlen - ein 'Zerrbild' des Maschinenprogramms.

Brüche pauken

Ein Lernprogramm unter Verwendung des ZX-81

Es zeigt sich immer wieder, daß gerade die Bruchrechnung bei Schülern auf wenig Gegenliebe stößt. Zum Teil liegt es daran, daß hier meißt reine Rechenfertigkeit abverlangt wird, die die Bewältigung einer großen Anzahl von Übungsaufgaben unvermeidlich macht.

Zur Aufgabenstellung und Kontrolle der einzelnen Lösungsschritte ist ein Rechner hervorragend geeignet.

Mit dem Programm 'Brüche pauken' kann die Addition und Subtraktion zweier Brüche umfassend geübt werden.

Der Bildschirm ist zweispaltig aufgebaut. Die linke Spalte ist dem Schüler vorbehalten, der dort die Zahlen an Ort und Stelle einträgt. In der rechten Spalte stellt der Rechner nach jedem Schritt seine Lösung gegenüber. Dies ermöglicht dem Schüler eine ständige Kontrolle. Weiß er nicht weiter oder möchte er einen Schritt auslassen, dann drückt er auf NEW LINE und der Rechner ist an der Reihe. Mit NEW LINE wird auch jede Zahleneingabe abgeschlossen.

Eine Aufgabe gliedert sich maximal in folgende Schritte: Aufgabenstellung, Kürzen der einzelnen Brüche (mit Wiederholmöglichkeit), Berechnung des Hauptnenners und Angabe der Zähler der erweiterten Brüche, Addition (bzw. Subtraktion) der Zähler, erneutes Kürzen, Umwandlung in gemischte Schreibweisse.

Ein Menü zu Beginn des Programmablaufs läßt ausser den eben beschriebenen Modus A noch die Wahl dreier anderer Modi zu.

B - der Rechner stellt die Aufgaben und führt die Lösung vor, ohne daß der Schülerer gefragt wird.

C - der Schüler stellt eigene Aufgaben, die er zusammen mit dem Rechner schrittweise löst.

D - Der Rechner führt die Lösung einer vom Schüler gestellten Aufgabe vor.

Den Schwierigkeitsgrad der Aufgabe stellt sich der Schüler selbst ein, indem er auf die Frage nach dem größtmöglichen Zähler und Nenner in der Aufgabenstellung die ihm genehme Antwort gibt.

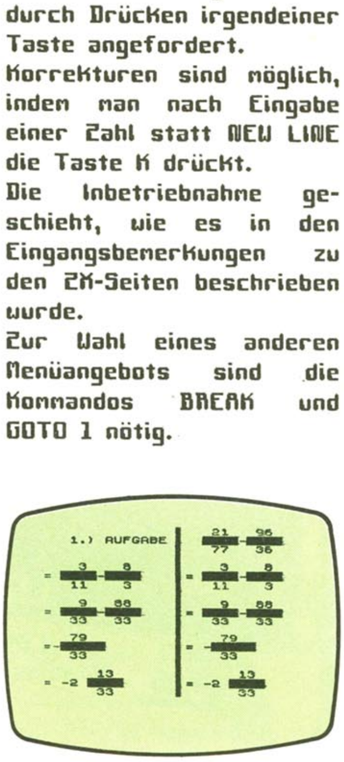
Diese obere Grenze für die Zahlen der Aufgabenstellung kann nicht über 99 und nicht unter 3 liegen. Trotzdem ergeben sich bei manchen Aufgaben während der Rechnung Zahlen über 10000.

Jede neue Aufgabe wird durch Drücken irgendeiner Taste angefordert.

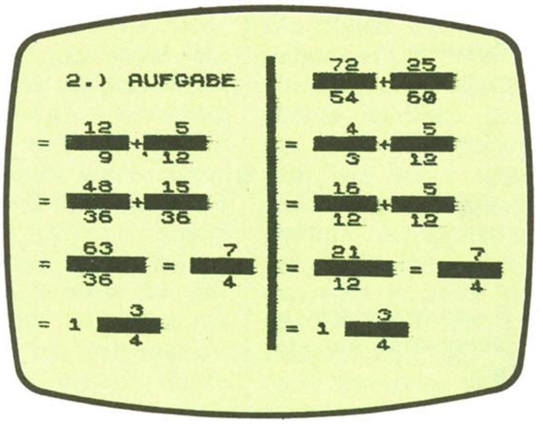
Korrekturen sind möglich, indem man nach Eingabe einer Zahl statt NEW LINE die Taste K drückt.

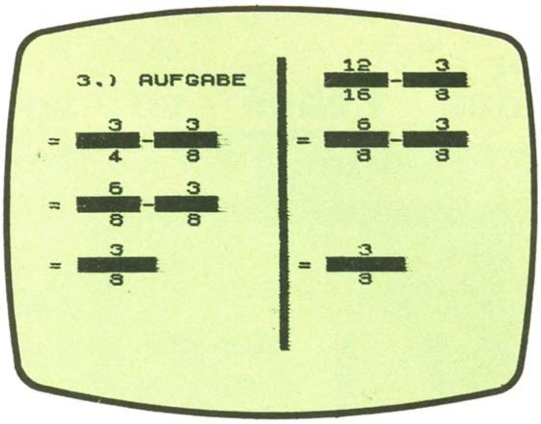
Die Inbetriebnahme geschieht, wie es in den Eingangsbemerkungen zu den ZX-Seiten beschrieben wurde.

Zur Wahl eines anderen Menüangebots sind die Kommandos BREAK und GOTO 1 nötig.




```
   2 GOTO 10
   3 CLEAR
   4 SAVE "B▒"
   5 GOSUB 900
   8 LET ANZ=0
  10 PRINT AT 11,1;"RECHNERAUFGA
BEN";AT 13,1;"RECHNERAUFGABEN -
MUSTER";AT 15,1;"SCHUELERAUFGABE
N";AT 17,1;"SCHUELERAUFGABEN - M
USTER"
  12 FOR M=-2 TO 1
  13 FOR J=1 TO 4
  14 PRINT AT 15+2*M,29;" "
  16 IF INKEY$<>"" THEN GOTO 21
  17 PRINT AT 15+2*M,29;"▒"
  18 NEXT J
  19 NEXT M
  20 GOTO 12
  23 CLS
  25 IF M<0 THEN GOTO 42
  26 CLS
  27 PRINT AT 20,0;"1.) ZAEHLER
?"
  28 INPUT Z1
  29 PRINT AT 20,0;"1.) NENNER
?"
  30 INPUT N1
  31 PRINT AT 20,0;"RECHENART ?
 "
  32 INPUT O$
  33 PRINT AT 20,0;"2.) ZAEHLER
?"
  34 INPUT Z2
  35 PRINT AT 20,0;"2.) NENNER
?"
  36 INPUT N2
  37 CLS
  41 GOTO 56
  42 PRINT AT 20,0;"GROESSTER ZA
EHLER ODER NENNER ?"
  43 INPUT C$
  44 FOR I=1 TO LEN C$
  45 IF C$(I)<"0" OR C$(I)>"9" T
HEN GOTO 43
  46 NEXT I
  47 IF VAL C$<3 OR VAL C$>99 TH
EN GOTO 43
  48 LET GZ=VAL C$
  49 CLS
  50 LET O$=CHR$ (21+INT (RND*2)
)
  51 LET Z1=INT (RND*GZ)+1
  52 LET Z2=INT (RND*GZ)+1
  53 LET N1=INT (RND*GZ)+1
  54 LET N2=INT (RND*GZ)+1
  55 IF INT (Z1/N1)=Z1/N1 OR INT
 (Z2/N2)=Z2/N2 THEN GOTO 50
  56 FOR I=2 TO 20
  57 PRINT AT I,15;"█"
  58 NEXT I
  61 LET J=0
  63 LET HN=1
  65 LET ANZ=ANZ+1
  66 FOR I=0 TO 21
  67 PRINT AT I,0;W$;AT I,16;W$
  70 NEXT I
  80 PRINT AT 3,2.5-LN ANZ*5;ANZ
;".) AUFGABE"
 100 GOSUB 800
 111 IF ABS M=1 THEN GOTO 168
 112 PRINT AT 21,0;"KUERZEN "
 122 PRINT AT 7,0;"= ";S$;O$;AT
6,2;V$;AT 8,2;V$
 129 LET PZ=6
 130 LET PK=5
 131 LET LS=1
 132 GOSUB 6000
 134 LET Z4=ZL
 135 IF CODE Z$<>118 OR ZL THEN
GOTO 138
 136 PRINT AT 7,0;W$
 137 GOTO 168
 138 LET J=1
 139 PRINT AT 6,2+(Z4<1000)+(Z4<
10);Z4
 140 LET PZ=8
 141 GOSUB 6000
 142 IF CODE Z$<>118 OR ZL THEN
GOTO 145
 143 PRINT AT 6,0;W$;AT 7,0;W$
 144 GOTO 168
 145 LET N4=ZL
 146 PRINT AT 8,4-LN N4/2*5;N4;A
T 7,7;S$;AT 6,7;U$;AT 8,7;U$
 147 LET PZ=6
 148 LET PK=10
 149 GOSUB 6000
 150 IF CODE Z$<>118 OR ZL THEN
GOTO 153
 151 PRINT AT 6,0;W$;AT 7,0;W$;A
T 8,0;W$
 152 GOTO 168
 153 LET Z5=ZL
 154 PRINT AT 6,9-LN Z5/2*5;Z5
 155 LET PZ=8
 156 GOSUB 6000
```

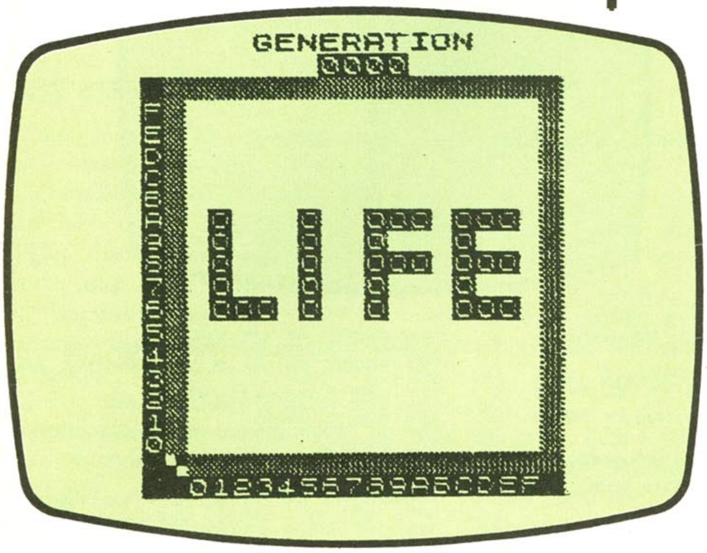


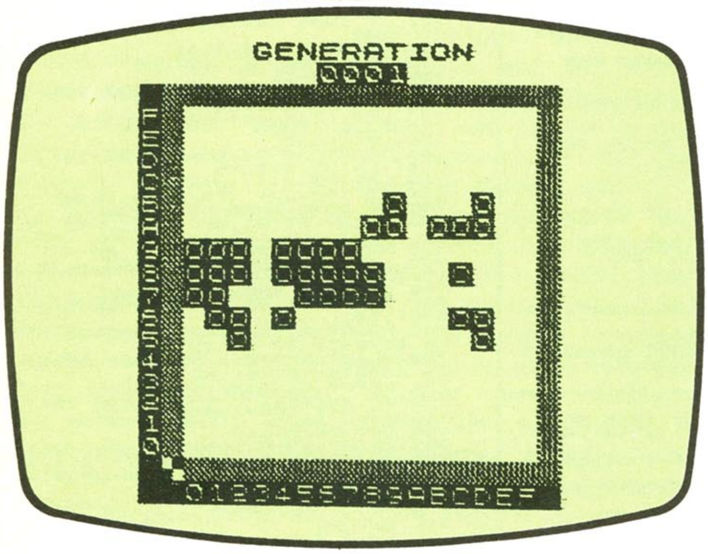



```
 157 IF CODE Z$<>118 OR ZL THEN GOTO 160
 158 PRINT AT 6,0;W$;AT 7,0;W$;AT 8,0;W$
 159 GOTO 168
 160 LET N5=ZL
 161 PRINT AT 8,9-LN N5/2*5;N5
 164 PRINT AT 21,0;"NOCH MAL KUERZEN? (J/N)"
 165 IF INKEY$="" THEN GOTO 165
 166 PRINT AT 21,0;W$+W$
 167 IF INKEY$="J" THEN GOTO 117
 169 LET KZ=1
 170 FOR I=1 TO 25
 171 IF Z1/P(I)<>INT (Z1/P(I)) OR N1/P(I)<>INT (N1/P(I)) THEN GOTO 190
 172 IF INT (N1/N2)=N1/N2 AND INT (INT (N1/N2)/P(I))<>INT (N1/N2)/P(I) THEN GOTO 190
 173 IF INT (N2/N1)=N2/N1 THEN GOTO 191
 174 LET KZ=KZ*P(I)
 175 LET Z1=Z1/P(I)
 176 LET N1=N1/P(I)
 177 PRINT AT 7,18;S$
 179 PRINT AT 21,0;"KUERZEN";AT 21,16;"KUERZUNGSZAHL ";KZ;AT 6,18;U$;AT 8,18;U$;AT 6,20-LN Z1/2*5;Z1
 185 PRINT AT 8,20-LN N1/2*5;N1;AT 21,16;W$+" "
 187 LET J=1
 188 GOTO 171
 190 IF INT (N1/N2)=N1/N2 THEN LET HN=N1
 191 IF P(I)>Z1 OR P(I)>N1 THEN GOTO 249
 192 NEXT I
 249 LET KZ=1
 250 FOR I=1 TO 25
 251 IF Z2/P(I)<>INT (Z2/P(I)) OR N2/P(I)<>INT (N2/P(I)) THEN GOTO 270
 252 IF INT (N2/N1)=N2/N1 AND INT (INT (N2/N1)/P(I))<>INT (N2/N1)/P(I) THEN GOTO 270
 253 IF INT (N1/N2)=N1/N2 THEN GOTO 271
 254 LET KZ=KZ*P(I)
 255 LET Z2=Z2/P(I)
 256 LET N2=N2/P(I)
 257 PRINT AT 7,23;S$
 259 PRINT AT 21,0;"KUERZEN";AT 21,16;"KUERZUNGSZAHL ";KZ;AT 6,23;U$;AT 8,23;U$;AT 6,25-LN Z2/2*5;Z2
 265 PRINT AT 8,25-LN N2/2*5;N2;AT 21,16;W$+" "
 267 LET J=1
 259 GOTO 251
 270 IF INT (N2/N1)=N2/N1 THEN LET HN=N2
 271 IF P(I)>Z2 OR P(I)>N2 THEN GOTO 273
 272 NEXT I
 273 PRINT AT 21,0;W$
 274 GOSUB 800
 300 IF HN<>1 OR N1=N2 THEN GOTO 370
 301 PRINT AT 21,0;"HAUPTNENNER"
 305 FOR I=1 TO 25
 307 LET POT=1
 310 FOR L=1 TO 7
 315 LET POT=POT*P(I)
 320 IF INT (N1/POT)-N1/POT=0 THEN GOTO 324
 322 IF INT (N2/POT)-N2/POT THEN GOTO 350
 324 LET HN=HN*P(I)
 326 IF HN-N1*N2=0 THEN GOTO 370
 330 NEXT L
 350 NEXT I
 370 LET J=J+1
 372 IF ABS M=1 THEN GOTO 407
 373 LET PZ=4+4*J
 374 LET PK=5
 375 PRINT AT PZ-1,2;S$;" ";S$;AT 21,0;"HAUPTNENNER"
 376 GOSUB 6000
 377 IF CODE Z$<>118 OR ZL THEN GOTO 380
 378 PRINT AT PZ-1,0;"
 "
 379 GOTO 407
 380 LET N4=ZL
 381 PRINT AT PZ-1,0;"=";TAB 6;O$;AT PZ,4-LN N4/2*5;N4;AT PZ,9-LN N4/2*5;N4
 386 LET PZ=2+4*J
 387 GOSUB 6000
 388 LET Z4=ZL
 389 IF CODE Z$<>118 OR ZL THEN GOTO 398
 390 PRINT AT PZ+1,0;W$;AT PZ+2,0;W$
 391 GOTO 407
 398 PRINT AT PZ,4-LN Z4/2*5;Z4
 399 LET PK=10
 400 GOSUB 6000
 402 LET Z5=ZL
 403 IF CODE Z$<>118 OR ZL THEN GOTO 406
 404 PRINT AT PZ+1,0;W$;AT PZ+2,0;W$
 405 GOTO 407
 406 PRINT AT PZ,9-LN Z5/2*5;Z5
 407 LET PZ=4+4*J
 408 IF N1=N2 THEN GOTO 610
 409 LET Z1=Z1*HN/N1
 410 LET N1=HN
 420 LET Z2=Z2*HN/N2
 430 LET N2=HN
 451 PRINT AT 21,16;AT PZ,20-LN N1/2*5;N1;TAB 25-LN N2/2*5;N2
 458 PRINT AT 21,0;W$;TAB 16;W$
 460 GOSUB 800
 610 IF ABS M=1 THEN GOTO 650
 611 LET PZ=2+4*(J+1)
 612 LET PK=6
 613 PRINT AT 21,0;W$;AT PZ+1,0;"= ";T$
 622 GOSUB 6000
 623 IF CODE Z$<>118 OR Z' THEN GOTO 626
 624 PRINT AT PZ+1,0;"          "
 625 GOTO 650
 626 LET Z6=ZL
 627 IF Z6<0 THEN PRINT AT PZ+1,1;"-"
 628 PRINT AT PZ,2+(ABS (Z6)<1000)+(ABS (Z6)<10);ABS (Z6)
 630 LET PZ=4+4*(J+1)
 635 GOSUB 6000
 636 LET N6=ZL
 638 PRINT AT PZ,2+(N6<10000)+(N6<10);N6
 650 LET PZ=3+4*(J+1)
 651 LET Z3=Z1+Z2-(CODE O$=22)*2*Z2
 652 LET N3=N2
 655 PRINT AT PZ,16;"= ";("-" AND Z3<0);T$
 659 PRINT AT PZ-1,18+(ABS (Z3)<10000)+(ABS (Z3)<10)+(Z3<0);ABS (Z3)
 660 PRINT AT PZ+1,18+(N3<10000)+(N3<10)+(Z3<0);N3
 661 IF ABS M=1 OR Z$="L" THEN GOTO 680
 662 PRINT AT 21,0;"KUERZEN"
 663 PRINT AT PZ,6;"= ";S$
 664 LET PZ=2+4*(J+1)
 665 LET PK=13
 666 GOSUB 6000
 667 IF CODE Z$<>118 OR ZL THEN GOTO 670
 668 PRINT AT PZ+1,6;"        "
 669 GOTO 680
 670 LET Z7=ZL
 671 IF Z7<0 THEN PRINT AT PZ+1,9;"-"
 672 PRINT AT PZ,12-LN ABS Z7/2*5;ABS Z7
 673 LET PZ=4+4*(J+1)
 674 GOSUB 6000
 675 IF CODE Z$<>118 OR ZL THEN GOTO 678
 676 PRINT AT PZ-1,8;"        ";AT PZ,8;"
 "
 677 GOTO 680
 678 LET N7=ZL
 679 PRINT AT PZ,10+(N7<10000)+(N7<10);N7
 680 LET PZ=3+4*(J+1)
 681 IF INT (Z3/N3)=Z3/N3 THEN GOTO 700
 682 LET KZ=1
 683 FOR I=1 TO 25
 684 IF Z3/P(I)<>INT (Z3/P(I)) OR N3/P(I)<>INT (N3/P(I)) THEN GOTO 698
 685 LET KZ=KZ*P(I)
 686 LET Z3=Z3/P(I)
 687 LET N3=N3/P(I)
 688 PRINT AT PZ,24;"=";TAB 26+(Z3<0);S$
 690 PRINT AT 21,16;"KUERZUNGSZAHL ";KZ
 691 PRINT AT PZ-1,26;U$
 692 PRINT AT PZ+1,26;V$
 693 IF Z3<0 THEN PRINT AT PZ,26;"-"
 694 PRINT AT PZ-1,28-LN ABS Z2/2*5+(Z3<0);ABS (Z3)
 695 PRINT AT PZ+1,28-LN N3/2*5+(Z3<0);N3
 696 PRINT AT 21,16;W$
 697 GOTO 681
 698 IF P(I)>Z3 OR P(I)>N3 THEN GOTO 700
 699 NEXT I
 700 PRINT AT 21,0;W$
 703 IF ABS M=1 THEN GOTO 733
 704 PRINT AT 21,0;"GEMISCHTE ZAHL"
 706 LET PZ=3+4*(J+2)
 708 LET PK=4
 710 GOSUB 6000
 712 IF CODE Z$<>118 OR ZL THEN GOTO 715
 713 PRINT AT PZ,0;V$
 714 GOTO 733
 715 LET G8=ZL
 718 PRINT AT PZ,0;"= ";G8;TAB 4+(ABS G8>10)+(G8<0);S$
 720 LET PZ=2+4*(J+2)
 721 LET PK=7+(ABS G8>10)+(G8<0)
 722 GOSUB 6000
 723 IF CODE Z$<>118 OR ZL THEN GOTO 726
 724 PRINT AT PZ+1,4;"             "
 725 GOTO 733
 726 LET Z8=ZL
 727 PRINT AT PZ,6+(ABS G8>10)+(G8<0)-LN Z8/2*5;Z8
 728 LET PZ=4+4*(J+2)
 729 GOSUB 6000
 730 LET N8=ZL
 731 PRINT AT PZ,6+(ABS G8>10)+(G8<0)-LN N8/2*5;N8
 733 LET PZ=3+4*(J+2)
 734 LET G3=INT (Z3/N3)+(Z3<0)
 735 IF G3=0 THEN GOTO 754
 736 PRINT AT 21,0;"GEMISCHTE ZAHL"
 737 PRINT AT PZ,16;"= ";G3
 739 LET Z3=(Z3-G3*N3)-(Z3<0)*2*(Z3-G3*N3)
 742 IF Z3=0 THEN GOTO 754
 745 PRINT AT PZ,20+(ABS G3>9)+(G3<0);S$
 748 PRINT AT PZ-1,22+(ABS G3>9)-LN Z3/2*5+(G3<0);Z3
 750 PRINT AT PZ+1,22+(ABS G3>9)-LN N3/2*5+(G3<0);N3
 754 PRINT AT 21,0;W$
 756 IF INKEY$="" THEN GOTO 756
 757 IF A<0 THEN GOTO 50
 760 GOTO 26
 800 LET PZ=3+4*J
 802 PRINT AT PZ,16;("=" AND J)
 804 PRINT AT PZ,18;S$;AT PZ-1,20-LN Z1/2*5;Z1;AT PZ+1,20-LN N1/2*5;N1;AT PZ,22;O$;S$
 806 PRINT AT PZ-1,25-LN Z2/2*5;Z2;AT PZ+1,25-LN N2/2*5;N2
 810 RETURN
 900 FAST
 901 RAND
 902 LET U$="     "
 903 LET V$="      "
 904 LET W$="                "
 905 LET Z$=""
 906 LET S=1/LN 10+.001
 907 LET VZ=1
 908 LET S$="█████"
 910 LET T$="██████"
 950 DIM P(25)
 952 LET P(1)=2
 954 LET K=1
 956 FOR I=3 TO 99 STEP 2
 958 FOR J=1 TO K
 960 IF INT (I/P(J))=I/P(J) THEN GOTO 968
 962 NEXT J
 964 LET K=K+1
 966 LET P(K)=I
 968 NEXT I
 997 SLOW
 999 RETURN
6005 LET ZL=0
6006 LET LR=0
6007 PRINT AT PZ,PK;" "
6008 PRINT AT PZ,PK;"▒"
6009 LET Z$=INKEY$
6010 IF Z$="" THEN GOTO 6007
6011 IF Z$<>"-" THEN GOTO 6017
6012 LET VZ=-1
6013 LET Z$="0"
6014 GOTO 6019
6015 IF Z$<"0" OR Z$>"9" THEN GOTO 6007
6017 IF Z$="K" OR CODE Z$=118 THEN GOTO 6025
6018 IF Z$<"0" OR Z$>"9" THEN GOTO 6007
6019 LET ZL=VZ*(ABS (ZL*10)+VAL Z$)
6020 LET LR=LR+1
6021 IF ZL THEN PRINT AT PZ,PK-LR;ZL
6022 IF ZL=0 THEN PRINT AT PZ,PK-LR;"-"
6023 GOTO 6007
6025 PRINT AT PZ,PK-4;V$
6027 IF Z$="K" THEN GOTO 6005
6029 LET VZ=1
6030 RETURN
```

LIFE

Wie das Leben so spielt





Life wurde von Conway auf der Suche nach einen Spiel erdacht, daß Lebensprozesse simuliert und die Auswirkungen von Isolation, Kooperation, Überbevölkerung usw. darstellen kann. Mit verblüffend einfachen Spielregeln entstand so ein Spiel mit äußerst reizvollen Mustern und überraschenden Verläufen.

Es steht ein großes Feld zur Verfügung (im Programm ein 16*16-Feld). Auf die einzelnen Zellen dieses Feldes werden die Mitglieder einer Gemeinschaft verteilt. Diese Gemeinschaft entwickelt sich nun von Generation zu Generationen nach folgenden Regeln:

Geburt - in einer freien Zelle entsteht ein neues Mitglied, wenn es genau drei Nachbarn hat.

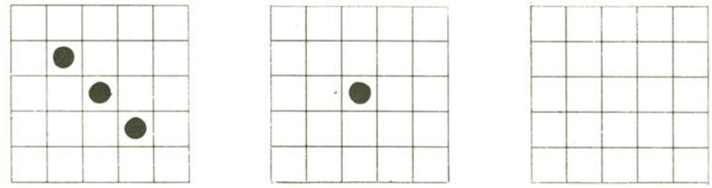
Tod - jedes Mitglied mit vier oder mehr Nachbarn stirbt an Übervölkerung; das gleiche Schicksal erleiden Mitglieder mit einen oder keinen Nachbarn wegen Isolation.

Überleben - kann ein Mitglied also nur dann, wenn es zwei oder drei Nachbarn hat.

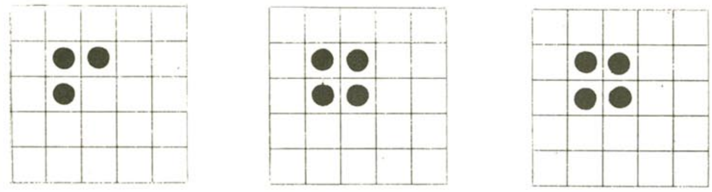
Als Nachbarschaft gelten säntliche acht angrenzenden Zellen.

Man kann die Schicksale folgender Figuren leicht nachvollziehen:

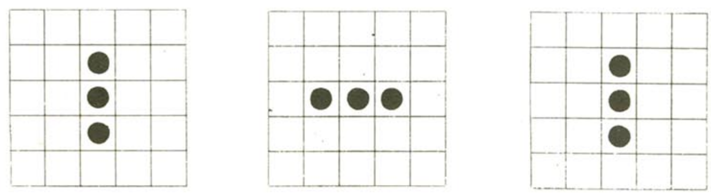
Figur I stirbt in der 2. Generation aus.



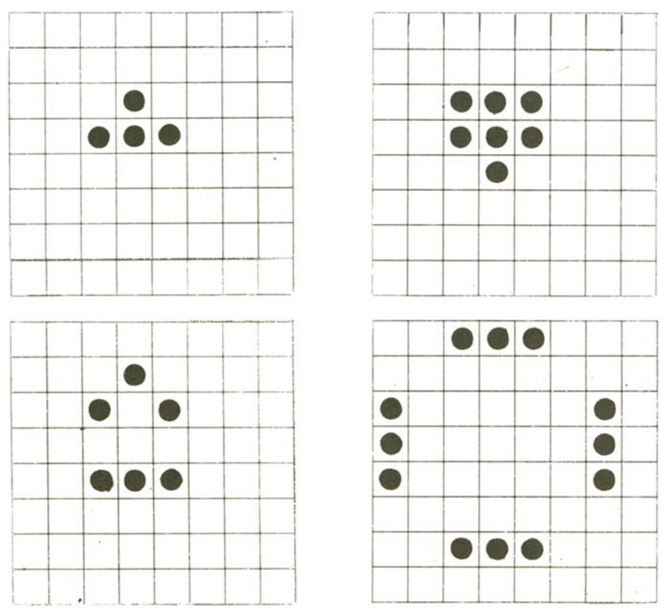
Figur II erweist sich ab der 1. Generation als stabiler Klotz.



Figur III ("Blinker" genannt) oszilliert bis in alle Ewigkeit.



Figur IV endet nach 10 Generationen als vierfacher Blinker.



Stabile Figuren gibt es übrigens in großer Zahl, während periodisch veränderliche (wie der Blinker) nicht so reich gesät sind. Wenn man eigene Figuren aufbaut und sie ihrem Conwayschen Schicksal überläßt, dann ist man immer wieder überrascht, welche Entwicklungen sich ergeben können. Vom Aussterben bedrohte Minderheiten erwachen plötzlich

zu neuen Leben. Es spalten sich Randgruppen ab, die in ihrem Außenseiterdasein recht beständig sind und alsbald von der großen Masse wieder integriert werden können. Und plötzlich fährt (ohne Flachs!) ein Auto daher; es fehlt nur noch der Zebrastreifen. Das Überraschendste ist, daß man einer Figur durch nichts ansehen kann, wie sie sich entwickeln wird, obwohl ihr Schicksal vollkommen vorherbestimmt ist.

Im 16*16-Feld muß jede Figur früher oder später aussterben, es sei denn, sie endet in einer oder mehrerer stabiler oder periodischer Figuren. Die in diesem Sinne längste von mir bisher beobachtete Entwicklungsdauer lag über 400 Generationen.

Das Programm erleichtert die Suche nach solchen Dauerbrennern dadurch, daß es Perioden bis zur Länge 6 erkennen kann und bei ihnen die Entwicklung stoppt. Perioden über mehr als 6 Generationen sind äußerst selten (oder sogar unmöglich?)

In einem von zwei möglichen Modi kann der Benutzer mit eigenen Figuren experimentieren. Ein Cursor wird mit den Richtungstasten 5 bis 8 bewegt. Taste P baut die Figur auf. Mit dem Cursor können auch einzelne Figurteile wieder gelöscht werden. NEW LINE startet den Entwicklungsprozeß. Durch Drücken irgendeiner Taste kann der Prozeß unterbrochen werden. Figuren mit besonders interessanten Schicksal lassen sich mit Taste W automatisch wieder auf das Feld stellen (dies klappt jedoch nur, wenn beim Aufbau keine Figurteile gelöscht wurden).

Man wird bald feststellen, daß viele Figuren stabil oder periodisch enden. Damit der Prozeß in diesen Fällen nicht unendlich weiterläuft, kann die Speicherzelle 16727 verändert werden. Sie legt die maximale Periodenlänge fest, die der Rechner gerade noch erkennt, woraufhin er abstoppt und die Periodenlänge unter dem Feld in einer Meldung mitteilt. POKE 16727,6 zum Beispiel läßt den Rechner noch Perioden der Länge 6 erkennen. Die gepokte Zahl darf nicht über 6 und nicht unter 1 liegen.

Das Legen von Anfangsfiguren können Sie jedoch auch dem Rechner selbst überlassen. Er verteilt ziemlich wahllos 30 bis 79 Mitglieder auf dem Feld. Soll er dies ununterbrochen tun, dann löschen Sie Zeile 2704. Der Moduswechsel von einer Betriebsart zur anderen ist in jeder Phase des Programms durch Drücken der Taste M möglich.

Zum Maschinenprogramm:
16514-16515: Zeiger im neusten Feld.
16516-16633: Das neue Feld wird berechnet.
16634-16601:Der Generationen-Zähler wird um eins erhöht und auf den Bildschirm geschrieben.
16662-16725: Das neue Feld wird auf den Bildschirm gedruckt.
16726-16795: Das neuste Feld wird mit den vergangenen verglichen, um Perioden zu entdecken.
16795: Periodenzähler
16796-: Das neuste und die vergangenen Felder.

Labels sind jeweils innerhalb von Befehlen durch ein vorangestelltes Q erkenntlich.

Die nächste Figur wird jeweils durch Drücken irgendeiner Taste angefordert.

Zur Inbetriebnahme des Programms sei auf die Eingangsbemerkungen zu den ZX-Seiten verwiesen.

```
   1 REM 580 ZEICHEN
   2 GOTO 10
   3 CLEAR
   4 SAVE "LIFE"
   5 GOSUB 2900
2010 FOR I=1 TO 16
2012 PRINT AT 19,7+I;"▒";AT 20,7
+I;CHR$ (155+I);AT 2+I,5;CHR$ (1
72-I);TAB 7;;"▒";TAB 24;"▒";AT 2
,7+I;"▒"
2014 NEXT I
2020 PRINT AT 2,6;"█▒";TAB 24;"▒
";AT 19,6;"█▀";TAB 24;"▒";AT 20,
6;"██";TAB 24;"█"
2030 PRINT AT 0,11;"GENERATION"
2060 PRINT AT 1,14;"0000"
2073 LET M=1
2074 LET C=0
2080 LET D$="8999A9B9C9CACB8D9DA
DBDCD8G9GAGBGCG8H8IAH8K9KAKBKCK8
L9MALAMCLCM"
2082 FOR I=1 TO LEN D$ STEP 2
2084 PRINT AT CODE D$(I)-28,CODE
 D$(I+1)-28;CHR$ 180
2086 NEXT I
2088 PRINT AT 15,9;"█ WIEDERHOLU
NG";AT 17,9;"█ MODUSWECHSEL"
2090 GOTO 2700
2100 LET D$=""
2150 IF M THEN GOTO 2400
2220 LET S=1
2221 IF CODE INKEY$ THEN GOTO 22
21
2223 LET Z=18
2224 LET K=8
2225 PRINT AT Z,K;"▒"
2226 LET C=CODE INKEY$
2227 IF NOT C THEN GOTO 2226
2230 IF C<>53 THEN GOTO 2240
2232 PRINT AT Z,K;CHR$ 180
2233 LET S=0
2236 LET D$=D$+CHR$ (Z+28)+CHR$
(K+28)
2239 GOTO 2226
2248 IF C=118 THEN GOTO 2500
2250 PRINT AT Z,K;(" " AND S)
2260 LET C=69-2*C
2270 LET Z=18.5-ABS (16-ABS (Z+(
ABS C=1)*C-2.5))
2280 LET K=23.5-ABS (16-ABS (K-(
ABS C=3)*C/3-7.5))
2290 PRINT AT Z,K;"▒"
2295 LET S=1
2300 GOTO 2226
2400 RAND
2402 FOR I=1 TO 30+INT (RND*50)
2405 LET Z=3+INT (RND*16)
2410 LET K=8+INT (RND*16)
2415 PRINT AT Z,K;CHR$ 180
2420 LET D$=D$+CHR$ (Z+28)+CHR$
(K+28)
2425 NEXT I
2500 IF CODE INKEY$ THEN GOTO 25
00
2510 RAND USR 16524
2550 LET C=CODE INKEY$
2600 IF NOT C AND NOT PEEK 16795
 THEN GOTO 2510
2610 LET I=PEEK 16795
2611 IF NOT I THEN GOTO 2700
2612 FOR J=0 TO 31
2614 IF PEEK (16796+J)<>255 THEN
 GOTO 2622
2616 NEXT J
2618 PRINT AT 21,10;"AUSGESTORBE
N"
2620 GOTO 2700
2622 FOR J=1 TO 3
2624 IF I>J THEN GOTO 2630
2626 PRINT AT 21,10;("    STABIL"
 AND J=1)+(STR$ J+"-ER PERIODE"
AND J>1)
2628 GOTO 2700
2630 NEXT J
2703 LET D=CODE INKEY$
2704 IF NOT C AND NOT D THEN GOT
O 2703
2705 FOR I=3 TO 18
2706 PRINT AT I,8;"
   "
2707 NEXT I
2708 PRINT AT 1,14;"0000";AT 21,
8;"
2710 IF C=50 THEN GOTO 2717
2711 IF C=60 THEN GOTO 2720
2712 IF C THEN GOTO 2100
2716 IF D<>50 THEN GOTO 2719
2717 LET M=1-M
2718 GOTO 2100
```

Die ZX-Seiten

```
2719 IF D<>60 THEN GOTO 2100
2720 FOR I=1 TO LEN D$ STEP 2
2722 PRINT AT CODE D$(I)-28,CODE
 D$(I+1)-28;CHR$ 180
2724 NEXT I
2730 GOTO 2500
2900 FAST
2990 LET C$="BC417EFEB420010C23C
9119C41ED5382402A0C40114A0019061
00E00111E00CD8440E5CD8440CD84401
9CD8440E523CD844019CD8440CD8440C
D8440E179FE032003B7180AFE0228033
718037EFE7FED5B82401A1712131A171
2E1108D13ED5382401111001 9E52A0C4
0115A0219EBE17BBD20A47ABC20A0119
C41ED5382402A0C4011330019060 37E3
CFEA62005369C2B10F5772A0C40116C0
0190610ED5B82408713 1ACB7F3EB4280
2373E00771B1A17121 31A17122310E41
3ED53824011110019E52A0C40117C021
9EBE17BBD20CB7ABC20C70603C548411
12000219C411910FD119C4106201ABE2
006231310F818030D20E479329B41C1C
5112000219C412B1910FDE519D1EBC14
80600CB21CB21CB21CB21CB21EDB8C9"
2992 FOR I=1 TO LEN C$/2
2993 POKE 16513+I,16*(CODE C$(2*
I-1)-28)+CODE C$(2*I)-28
2994 NEXT I
2997 SLOW
2999 RETURN
```

```
16514 00           15 NOP
16515 00              NOP
16516 7E           01 LD A,(HL)
16517 FE80            CP 180
16519 2001            JR NZ,1
16521 0C              INC C
16522 23              INC HL
16523 C9              RET
16524 119C41          LD DE,017
16527 ED538240        LD (015),DE
16531 2A0C40          LD HL,(16396)
16534 114A00          LD DE,74
16537 19              ADD HL,DE
16538 0610         16 LD B,16
16540 0E00         02 LD C,0
16542 111E00          LD DE,30
16545 CD8440          CALL 001
16548 E5              PUSH HL
16549 CD8440          CALL 001
16552 CD8440          CALL 001
16555 19              ADD HL,DE
16556 CD8440          CALL 001
16559 E5              PUSH HL
16560 23              INC HL
16561 CD8440          CALL 001
16564 19              ADD HL,DE
16565 CD8440          CALL 001
16568 CD8440          CALL 001
16571 CD8440          CALL 001
16574 E1              POP HL
16575 79              LD A,C
16576 FE03            CP 3
16578 2003            JR NZ,3
16580 B7              OR A
16581 180A            JR 10
16583 FE02            CP 2
16585 2803            JR Z,3
16587 37              SCF
16588 1803            JR 3
16590 7E              LD A,(HL)
16591 FE7F            CP 127
16593 ED5B8240        LD DE,(015)
16597 1A              LD A,(DE)
16598 17              RLA
16599 12              LD (DE),A
16600 13              INC DE
16601 1A              LD A,(DE)
16602 17              RLA
16603 12              LD (DE),A
16604 E1              POP HL
16605 10BD            DJNZ 002
16607 13              INC DE
16608 ED538240        LD (015),DE
16612 111100          LD DE,17
16615 19              ADD HL,DE
16616 E5              PUSH HL
16617 2A0C40          LD HL,(16396)
16620 115A02          LD DE,602
16623 19              ADD HL,DE
16624 EB              EX DE,HL
16625 E1              POP HL
16626 7B              LD A,E
16627 BD              CP L
16628 20A4            JR NZ,016
16630 7A              LD A,D
16631 BC              CP H
16632 20A0            JR NZ,016
16634 119C41          LD DE,017
16637 ED538240        LD (015),DE
16641 2A0C40          LD HL,(16396)
16644 113300          LD DE,51
16647 19              ADD HL,DE
16648 0603            LD B,3
16650 7E           12 LD A,(HL)
16651 3C              INC A
16652 FEA6            CP 166
16654 2005            JR NZ,5
16656 369C            LD (HL),156
16658 2B              DEC HL
16659 10F5            DJNZ 012
16661 77              LD (HL),A
16662 2A0C40       05 LD HL,(16396)
16665 116C00          LD DE,108
16668 19              ADD HL,DE
16669 0610         03 LD B,16
16671 ED5B8240     04 LD DE,(015)
16675 B7              OR A
16676 13              INC DE
16677 1A              LD A,(DE)
16678 CB7F            BIT 7,A
16680 3E80            LD A,180
16682 2802            JR Z,2
16684 37              SCF
16685 3E00            LD A,0
16687 77              LD (HL),A
16688 1B              DEC DE
16689 1A              LD A,(DE)
16690 17              RLA
16691 12              LD (DE),A
16692 13              INC DE
16693 1A              LD A,(DE)
16694 17              RLA
16695 12              LD (DE),A
16696 23              INC HL
16697 10E4            DJNZ 004
16699 13              INC DE
16700 ED538240        LD (015),DE
16704 111100          LD DE,17
16707 19              ADD HL,DE
16708 E5              PUSH HL
16709 2A0C40          LD HL,(16396)
16712 117C02          LD DE,636
16715 19              ADD HL,DE
16716 EB              EX DE,HL
16717 E1              POP HL
16718 7B              LD A,E
16719 BD              CP L
16720 20CB            JR NZ,003
16722 7A              LD A,D
16723 BC              CP H
16724 20C7            JR NZ,003
16726 0603            LD B,1
16728 C5              PUSH BC
16729 48              LD C,B
16730 41           06 LD B,C
16731 112000          LD DE,32
16734 219C41          LD HL,017
16737 19              ADD HL,DE
16738 10FD            DJNZ 253
16740 119C41          LD DE,017
16743 0620            LD B,32
16745 1A           07 LD A,(DE)
16746 BE              CP (HL)
16747 2006            JR NZ,008
16749 23              INC HL
16750 13              INC DE
16751 10F8            DJNZ 007
16753 1803            JR 009
16755 0D           08 DEC C
16756 20E4            JR NZ,006
16758 79           09 LD A,C
16759 329B41          LD (018),A
16762 C1              POP BC
16763 C5              PUSH BC
16764 112000          LD DE,32
16767 219C41          LD HL,017
16770 2B              DEC HL
16771 19              ADD HL,DE
16772 10FD            DJNZ 253
16774 E5              PUSH HL
16775 19              ADD HL,DE
16776 D1              POP DE
16777 EB              EX DE,HL
16778 C1              POP BC
16779 48              LD C,B
16780 0600            LD B,0
16782 CB21            SLA C
16784 CB21            SLA C
16786 CB21            SLA C
16788 CB21            SLA C
16790 CB21            SLA C
16792 EDB8            LDDR
16794 C9              RET
16795 00           18 NOP
16796 00           17 NOP
```

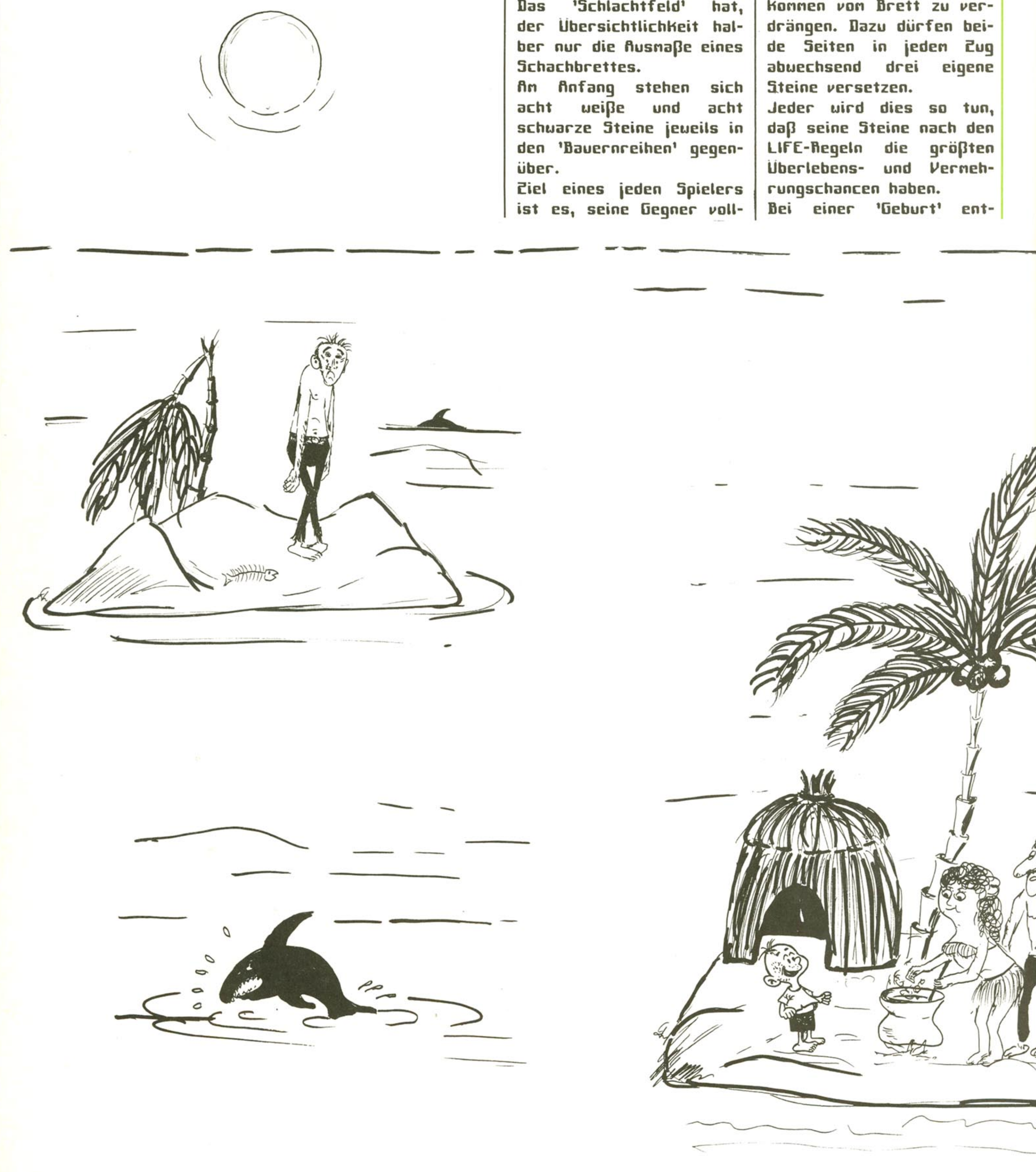


The Explorers Guide to the ZX
Mike Lord
Timedata
Programming Aid
Some Games and Other Novelties.
Applications.
Machine Language.
Discovering the ROM.
Hardware.
Edition 2
WAKE UP YOUR ATOM!
BRIAN LLOYD

Survive -

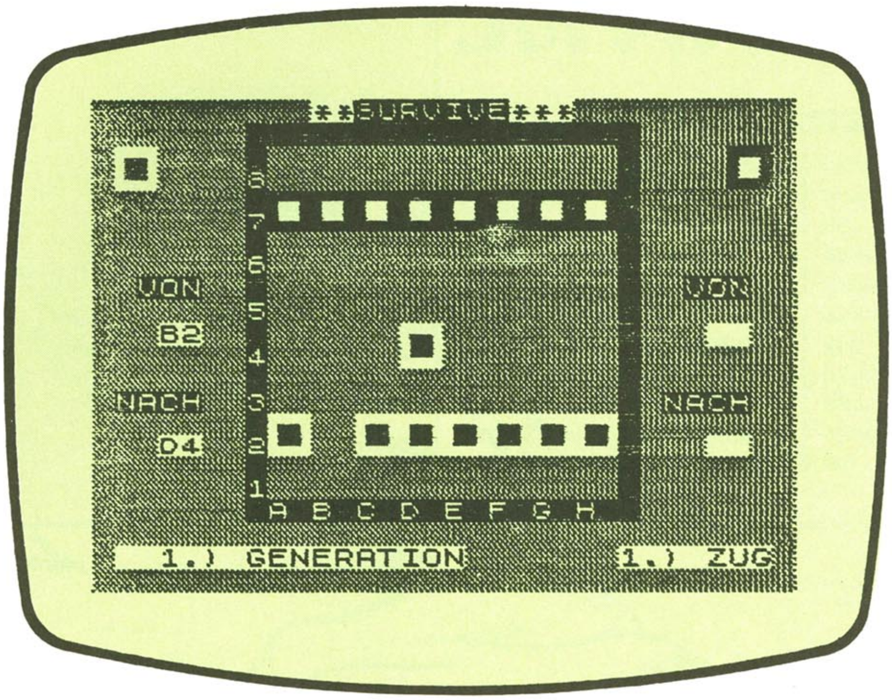
der Kampf ums Überleben

Das 'Schlachtfeld' hat, der Übersichtlichkeit halber nur die Ausmaße eines Schachbrettes.
Am Anfang stehen sich acht weiße und acht schwarze Steine jeweils in den 'Bauernreihen' gegenüber.
Ziel eines jeden Spielers ist es, seine Gegner vollkommen vom Brett zu verdrängen. Dazu dürfen beide Seiten in jeden Zug abwechsend drei eigene Steine versetzen.
Jeder wird dies so tun, daß seine Steine nach den LIFE-Regeln die größten Überlebens- und Vermehrungschancen haben.
Bei einer 'Geburt' ent-





scheidet die Übermacht in der Nachbarschaft darüber, welcher Seite der 'Neugeborene' angehört. Wenn z.B. ein freies Feld zwei scharze und einen weißen Nachbarn hat, dann kommt ein schwarzer Neuer hinzu.
Während das Maschinenprogramm zu LIFE mit 1 Bit pro Feldelement auskommt, arbeitet dasjenige von SURVIVE wegen dreier möglicher Zustände (leer - Spieler 1 - Spieler 2) mit 1 Byte pro Element.
Auch bei der Inbetriebnahme von SURVIVE gelten die Regeln, die in den Eingangsbemerkungen zu den ZX-Seiten nachzulesen sind.




```
   1 REM 888 ZEICHEN
   2█GOTO 100
   3 CLEAR
   4 SAVE "SUR"
   5 GOSUB 900
  10 DIM B(2)
 100 FOR I=16753 TO 16816
 101 POKE I,136
 102 PRINT "▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒";
 103 NEXT I
 130 PRINT AT 0,10;"**SURVIVE***
";TAB 7;"██████████████████"
 140 PRINT TAB 1;Z$(1,1);TAB 29;
Z$(2,1);TAB 1;Z$(1,2);TAB 29;Z$(
2,2)
 150 FOR I=1 TO 8
 152 PRINT AT 18,5+2*I;"█";CHR$
(165+I);"██"
 154 PRINT AT 18-2*I,7;"█";TAB 2
4;"█";TAB 7;CHR$ (156+I);TAB 24;
"█"
 156 NEXT I
 160 PRINT AT 8,2;"VON";TAB 27;"
VON";AT 13,1;"NACH";TAB 26;"NACH
"
 170 PRINT AT 20,1;"    .) GENERA
TION";TAB 24;" .) ZUG"
 210 FOR I=1 TO 2
 211 FOR J=1 TO 8
 212 PRINT AT 24-10*I,6+2*J;Z$(I
,1);TAB 6+2*J;Z$(I,2)
 213 NEXT J
 214 NEXT I
 220 FOR H=1 TO 99
 221 PRINT AT 10,3;"  ";TAB 28;"
  ";AT 15,3;"  ";TAB 28;"  "
 222 PRINT AT 20,3-(H>9)-(H>99);
H
 230 FOR J=1 TO 3
 232 PRINT AT 20,24;CHR$ (J+28)
 233 FOR R=1 TO 2
 234 PRINT AT 10,25*R-22;"  ";AT
15,25*R-22;"  "
 235 LET K=R+(INT (H/2)=H/2)
 236 IF K=3 THEN LET K=1
 237 FOR L=1 TO 2
 238 LET A=5+5*L
 239 FOR M=1 TO 2
 240 LET B=M+25*K-23
 241 PRINT AT A,B;"█"
 242 LET C=CODE INKEY$
 243 PRINT AT A,B;" "+(" " AND M
=1)
 244 IF NOT C THEN GOTO 241
 245 IF C=118 THEN GOTO 276
 246 IF C<47-9*M OR C>54-9*M THE
N GOTO 241
 252 LET D$(L,M)=CHR$ C
 253 PRINT AT A,B;D$(L,M)
 255 IF CODE INKEY$ THEN GOTO 25
5
 260 NEXT M
 262 IF PEEK (PEEK 16396+256*PEE
K 16397-66*CODE D$(L,2)+2*CODE D
$(L,1)+2375)<>(263-128*K AND L=1
)+(136 AND L=2) THEN GOTO 239
 265 NEXT L
 270 LET A=2*CODE D$(1,1)-68
 271 LET B=74-2*CODE D$(1,2)
 272 LET C=2*CODE D$(2,1)-68
 273 LET D=74-2*CODE D$(2,2)
 274 PRINT AT B,A;"▒▒";TAB A;"▒▒
";AT D,C;Z$(K,1);TAB C;Z$(K,2)
 276 IF CODE INKEY$ THEN GOTO 27
6
 277 NEXT R
 280 NEXT J
 466 LET J=RND**RND**RND
 475 RAND USR 16537
 480 IF PEEK 16448 AND PEEK 1644
9 THEN GOTO 600
 505 LET A=(PEEK 16448<=PEEK 164
49)+1
 506 LET B=28*A-27
 507 LET B(A)=B(A)+1
 508 PRINT AT 1,14;CHR$ (B(1)+15
6);"█";CHR$ (B(2)+156)
 510 FOR I=1 TO 40
 512 PRINT AT 2,B;"  ";TAB B;"
";AT 2,B;Z$(A,1);TAB B;Z$(A,2)
 514 NEXT I
 520 CLS
 530 GOTO 100
 600 NEXT H
 610 GOTO 505
 900 FAST
 920 DIM D$(2,2)
 930 DIM Z$(2,2,2)
 932 LET Z$(1,1)=CHR$ 135+CHR$ 4
 933 LET Z$(1,2)=CHR$ 2+CHR$ 1
 934 LET Z$(2,1)=CHR$ 7+CHR$ 132
 935 LET Z$(2,2)=CHR$ 130+CHR$ 1
29
 990 LET C$="ED5B3E407EFE8720021
41CFE07200114ED533E402323C911714
1ED533C40110000ED5340402A0C40110
700190E080608110000ED533E40CD824
0E5CD8240CD8240113C0019CD8240E52
323CD8240113C0019CD8240CD8240CD8
240E17AFE0320107EFE8820137BFE023
E07380C3E87180B7AFE023E8820017EE
D5B3C401213ED533C40E110AA1132001
90D20A1117141ED533C402A0C4011480
0190E080608ED5B3C401A13ED533C402
323E5FE882006118888D51822FE87201
0110201D516045F3A40403C324040180
E118281D516845F3A41403C324140287
22B7311210019D1732372E1108811320
0190D20AFC9"
 992 FOR J=1 TO LEN C$/2
 993 POKE 16513+J,16*(CODE C$(2*
J-1)-28)+CODE C$(2*J)-28
 994 NEXT J
 997 SLOW
 999 RETURN
```

Labyrinth-Strategien

Irrgärten spielten schon in den Mythen der Geschichte eine besondere Rolle. Chronisten und Dichter konnten sich ihrer Faszination nicht entziehen und überlieferten uns unzählige Legenden, Sagen und Märchen in denen Labyrinthe eine wichtige Rolle spielen.

Auch in unserer Zeit haben sich wieder viele Leute mit Labyrinthen beschäftigt. Es entstanden unter Zuhilfenahme von Computern Spielereien, in denen Labyrinthe eine dominierende Rolle einnehmen. Nicht nur Abenteuerspiele, sondern auch Denk- und Actionspiele greifen auf Irrgärten zurück.

Pacman - das wohl bekannteste dieser Spiele fasziniert Millionen von Video-Spielern in aller Welt.

Homecomputer möchte Ihnen in dieser Ausgabe zwei Strategien zur Erzeugung von Irrgärten vorstellen.

Jeder dieser Strategien schließt sich ein Programmbeispiel an, das sich leicht in die jeweilige Maschinensprache Ihres Microcomputers übersetzen läßt. Wenn die Routinen zum Generieren der Irrgärten in Maschinencode geschrieben werden, verkürzt sich die Ausführungszeit auf weniger als 1 % der Zeit, die ein Basic-Programm benötigen würde.

Die Wandstrategie

Im Gegensatz zur Wegstrategie werden, wie der Name schon sagt, die Wände aufgebaut.

Man stelle sich ein rechteckiges Feld wie ein Stück Rechenpapier mit quadratischen Kästchen gerastert vor. Die ersten vier Wände werden durch die Umrahmung vorgegeben. Auf den Linien des Rahmens werden im Laufe der Entstehung weitere Wände durch Aneinandersetzen kleiner Teilstücke zusammengesetzt.

Zum Schluß bekommt der Irrgarten einen Eingang und einen Ausgang, indem aus dem Rahmen zwei Teilstücke wieder herausgebrochen werden.

Der Algorithmus für eine Wand ist wie folgt angelegt: Suche einen Knotenpunkt des Rasters, der noch von keiner Wand belegt ist. Wähle einen Knotenpunkt in der unmittelbaren Nachbarschaft und ziehe zwischen beiden eine Verbindung, die ein Teilstück der neuen Wand darstellt. Setze dies fort, bis eine Verbindung zu einer schon bestehenden Wand festgestellt wird. Wenn auf diese Weise die Wand mit sich selbst in Kontakt kommt, wird sie wieder eingerissen. Im Programm wird sie allerdings gar nicht erst aufgebaut.

Diese Bedingung sorgt dafür, daß es von einer Stelle zur anderen eines auf diese Weise erzeugten Labyrinths nur einen Weg gibt und daß isolierte Gebiete, die unmauert sind nicht auftreten können.

So entstehen normalerweise am Anfang der Aufbauarbeiten, wenn noch genügend Platz ist, die längsten Wände, weil noch genügend Platz ist, während sie kurz vor Vollendung meist nur aus einigen Teilstücken bestehen.

Einen Irrgarten, in dem es mehrere Wege zum Ausgang geben kann, erreicht man durch Beschänkung der Wandlängen. Dabei werden nämlich weitere Schlupflöcher zugelassen, denn der Anschluß an eine Wand wird nicht immer vollzogen.

Der ganze Prozeß läßt sich auf dem Bildschirm beim Ausführen eines Basic-Programms gut verfolgen.

Programm zur Wandstrategie

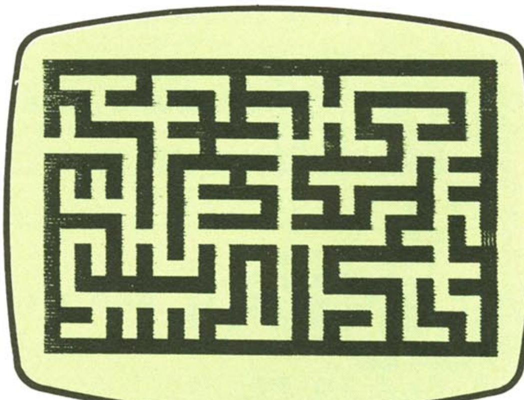
In Zeile 159 des Programms wird der Rechner veranlaßt, gegen Ende der Erzeugung den Wandbeginn anders festzulegen, als zuvor.

Zunächst baut er die Wand W auf, indem er im Feld G ständig nachschaut, ob auf der Stelle, die er zufällig gewählt hat, schon eine Wand ist oder nicht (0 oder 1). Zum Schluß wird ihm dies aufgrund der häufiger werdenden Mißerfolge zu frustrierend und er macht sich an die Arbeit, alle freien Stellen des Irrgartens in einer Reihe F abzulegen. Im weiteren Verlauf wirft er die jeweils neu bebauten Stellen aus dieser Reihe heraus. S hat die Funktion eines Schalters (0-1) zwischen beiden Arten des Suchens.

Auch dieses Programm enthält ein Unterprogramm in Maschinensprache und wird, wie in den Eingangsbemerkungen beschieben vorbereitet.

Nach der Generierung wird zu einem Rennen durch den Irrgarten aufgefordert. Die Tasten 5 - 8 des ZX-81 fungieren als Richtungstasten.

Ein neues Labyrinth wird durch Drücken irgend einer Taste erzeugt. Es gilt, die bisherigen Rekorde zu schlagen. Eine mitlaufende Uhr informiert ständig über die verstreichende Zeit.



Labyrinthe

```
   1 REM 390 ZEICHEN
   2 GOTO 10
   3 CLEAR
   4 SAVE "IGR"
   5 GOSUB 900
  11 RAND
  12 CLS
  13 DIM G(X+1,Y+1)
  14 LET S=0
  15 FOR I=1 TO X+1
  16 LET G(I,1)=1
  20 LET G(I,Y+1)=1
  21 NEXT I
  22 FOR I=1 TO Y+1
  23 LET G(1,I)=1
  24 LET G(X+1,I)=1
  25 NEXT I
  27 LET FP=(X-1)*(Y-1)
  28 FOR I=2 TO 2*(X+1)
  29 PRINT AT 2,I;"■"
  30 PRINT AT 2*(Y+1),I;"■"
  31 NEXT I
  32 FOR J=2 TO 2*(Y+1)
  33 PRINT AT J,2;"■"
  34 PRINT AT J,2*(X+1);"■"
  35 NEXT J
  50 LET WP=1
  52 DIM X(X+1,Y+1)
  53 IF S=1 THEN GOTO 65
  55 LET ZX=INT (RND*(X-1))+2
  56 LET ZY=INT (RND*(Y-1))+2
  58 IF G(ZX,ZY) THEN GOTO 55
  59 LET W(1,1)=ZX
  60 LET W(1,2)=ZY
  61 GOTO 70
  65 LET Z=INT (RND*FP)+1
  66 LET W(1,1)=F(Z,1)
  67 LET W(1,2)=F(Z,2)
  70 LET X(W(1,1),W(1,2))=1
  75 LET WP=WP+1
  80 LET XX=W(WP-1,1)
  85 LET YY=W(WP-1,2)
  90 IF X(XX+1,YY) AND X(XX-1,YY
) AND X(XX,YY+1) AND X(XX,YY-1)
THEN GOTO 50
 100 LET Z=INT (RND*4)+1
 105 LET W(WP,1)=W(WP-1,1)+R(Z,1
)
 110 LET W(WP,2)=W(WP-1,2)+R(Z,2
)
 115 LET XX=W(WP,1)
 120 LET YY=W(WP,2)
 125 IF X(XX,YY) THEN GOTO 80
 130 LET X(XX,YY)=1
 135 IF NOT G(XX,YY) THEN GOTO 7
5
 150 FOR I=1 TO WP
 154 LET XX=W(I,1)
 155 LET YY=W(I,2)
 157 LET G(XX,YY)=1
 158 NEXT I
 159 IF FP>(X-1)*(Y-1)*.1 THEN G
OTO 196
 160 IF S=1 THEN GOTO 171
 161 LET T=0
 162 FOR I=2 TO X
 163 FOR J=2 TO Y
 164 IF G(I,J) THEN GOTO 168
 165 LET T=T+1
 166 LET F(T,1)=I
 167 LET F(T,2)=J
 168 NEXT J
 169 NEXT I
 170 LET S=1
 171 FOR I=1 TO WP
 172 FOR J=1 TO FP
 173 IF F(J,1)=W(I,1) AND F(J,2)
=W(I,2) THEN GOTO 176
 174 NEXT J
 175 GOTO 196
 178 FOR K=J TO FP
 180 LET F(K,1)=F(K+1,1)
 182 LET F(K,2)=F(K+1,2)
 184 NEXT K
 195 NEXT I
 198 LET FP=FP-WP+1
 200 FOR I=1 TO WP-1
 201 LET XX=W(I,1)
 202 LET YY=W(I,2)
 203 PRINT AT 2*YY,2*XX;"■"
 204 PRINT AT YY+W(I+1,2),XX+W(I
+1,1);"■"
 212 NEXT I
 300 IF FP THEN GOTO 50
 305 LET Z=2*(INT (RND*(Y-1))+1)
+1
 306 LET S=2
 311 PRINT AT Z,S;" "
 315 LET Z=2*(INT (RND*(Y-1))+1)
+1
 316 LET S=2*(X+1)
 321 PRINT AT Z,S;"▒"
 380 SLOW
 385 PRINT AT 0,25;"PUNKTE"
 390 POKE 16514,Z
 392 POKE 16515,S
 395 POKE 16684,0
 396 POKE 16685,0
 397 POKE 16686,0
 398 POKE 16687,0
 400 RAND USR 16560
 403 IF PEEK 16515=1 THEN GOTO 4
11
 405 GOTO 400
 411 LET P=PEEK 16684+256*PEEK 1
6685+65536*PEEK 16686
 421 IF P>REK THEN GOTO 426
 422 FOR I=1 TO 20
 423 PRINT AT 0,2;"REKORD";AT 0,
2;"REKORD"
 424 NEXT I
 425 LET REK=P
 426 PRINT AT 0,21.5-LN P/LN 10;
P
 428 LET N=RND**RND**RND
 429 IF INKEY$="" THEN GOTO 429
 430 GOTO 10
 900 FAST
 902 LET REK=5000
 903 LET X=14
 904 LET Y=9
 905 DIM F((X+1)*(Y+1),2)
 907 DIM W((X+1)*(Y+1),2)
 910 DIM R(4,2)
 911 LET R(1,1)=-1
 912 LET R(2,1)=1
 913 LET R(3,2)=-1
 914 LET R(4,2)=1
 990 LET C$="000000002E0016000F0
62930011910FAC900002A92401E21CD5
640ED5B9440160019ED5B0C401911010
0195600C9ED5B8240ED538440CD4541C
D6B023EFFBC2004BD2001C9ED4B82407
C2F95FE29200105FE3120010CFE21200
10DFE192001043E0032AE40ED438240E
D439340CD95403A8340FE1F280AFE01C
83E80BA28021809ED5B8440ED538240C
93E7732AE40ED4B8440ED4393403E00C
D9540ED4B8240ED4393403E85CD9540C
9000000002100000610CB1117ED6AED5
23001193F10F3CB11C9ED5B2C4113ED5
32C413E00BA200CBB2009ED5B2E4113E
D532E41ED4B2C417811E803CD30414D7
C116400CD3041452A0C40111400193E9
C8177483E00110A00CD3041452A0C401
11500193E9C8177233E9C8077C9"
 992 FOR I=1 TO LEN C$/2
 993 POKE 16513+I,16*(CODE C$(2*
I-1)-28)+CODE C$(2*I)-28
 994 NEXT I
 997 SLOW
 999 RETURN
```

Labyrinthe

Die Wegstrategie

Diese Methode der Labyrintherzeugung geht davon aus, daß ein Gebiet mit Räumen bebaut ist und die Wände dieser Räume schon stehen.
Der Rechner hat hier die Aufgabe, Wände einzureissen und bahnt sich auf diese Weise Wege, die schließlich einen reizvollen Irrgarten ergeben.
Um dem Computer plausibel zu machen, wo im Labyrinth Wände stehen und wo nicht, müssen Zahlenwerte benutzt werden, denn Zahlen sind das Einzige, womit er etwas anfangen kann.
Weil eine Zelle 0, 1, 2, 3 oder 4 Wände haben kann, haben Mathematiker die sich lange vor uns mit Irrgärten beschäftigt haben, Möglichkeiten ersonnen, die verschiedenen vorkommenden Konstellationen mit Zahlen auszudrücken.
Wieder einmal waren Primzahlen der Schlüssel zum Erfolg. Unsere Wegstrategie macht sich diese Methode zunutze.
Die Wände im Norden, Westen, Osten und Süden erhalten die Bezeichnungen 2, 3, 5 und 7. Besitzt eine Zelle mehrere Wände, so werden einfach die entsprechenden Zahlen miteinander multipliziert und das Ergebnis deutet unmißverständlich die Positionen der Wände an. Eine Zelle, in der die Wand im Westen fehlt, wird also mit 2*5*7=70 gespeichert.
Auf diese Weise ist es leicht, den Computer beizubringen, Labyrinthe zu erzeugen. Zu Beginn haben alle Zellen 4 Wände. Demnach befinden sich in allen Speicherzellen des Rechners, in denen das Labyrinth abgelegt wird, die Zahl 210, denn 2*3*5*7=210.
Auf seinen bahnbrechenden Weg muß der Computer sich immer durch Doppelwände arbeiten. Einfache Wände durchbricht er nicht! Deshalb wird die am Anfang durch Nullen dargestellte Umrandung nie von ihm "betreten", und er wird auch niemals eine Wand zu einer schon einmal durchlaufenen Zelle durchbrechen - was garantiert, daß es am Ende nur einen möglichen Weg durch das Labyrinth gibt.

Programm zur Wegstrategie

Der Algorithmus hierzu sieht so aus:
Generiere Zufallszahl 0 bis 3 für die Marschrichtung.
Teste, ob die Zelle in der entsprechenden Richtung gleich 210 ist.
Wenn nein, dann teste, ob es sich um eine Sackgasse handelt; ansonsten versuche eine andere Richtung mit neuer Zufallszahl.
Wenn ja, dann durchbreche die Wand; was wie folgt geschieht: Angenommen, der Weg geht nach Süden. Die Zahl in der aktuellen Zelle wird dann durch 7 geteilt und die Zahl in der neuen Zelle durch 2.
Erkennt der Rechner, daß er sich in einer Sackgasse befindet, so stehen ihm zwei Fortsetzungen zur Verfügung.
Die eine: Er "merkt" sich ohnehin, welchen Weg er geht (Array mit Positions-Koordinaten) und springt zufällig an eine schon betretene Stelle. Das passiert immer vom Beginn an solange, bis eine vorher bestimmte Anzahl von Zellen zerstört wurde.
Danach ändert sich die Methode: Es geht den soeben beschrittenen Weg Schritt für Schritt zurück, bis eine Nachbarzelle gefunden wird, die die Zahl 210 enthält. Die Wände zu dieser Zelle werden durchbrochen und der Weg (wie oben) mittels Zufallszahlen fortgesetzt.

Das Muster-Programm zur Wegstrategie wurde für den Apple II geschrieben.

Der Rechner fragt nach dem Starten nach den Dimensionen des Labyrinths. Danach zeichnet er das Grundmuster mit allen Wänden und beginnt sich seine Wege zu bahnen. Im Basic ist dieser Vorgang schön zu verfolgen.

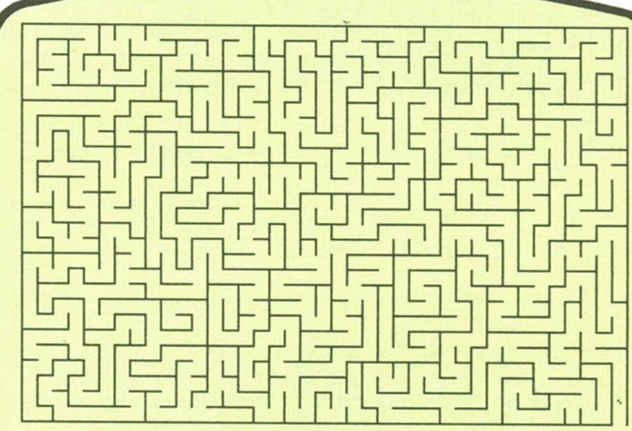
```
ÜLIST

100  HIMEM: 16384
200  TEXT : HOME : INPUT "DIMENSIONEN (39,26) ? ";DX,DY:DX = DX - 1:DY = DY - 1:DI = DX * DY
300  DIM F%(38,25),NX%(1000),NY%(1000),XX%(3),YY%(3),D%(3)
400  FOR I = 0 TO DX: FOR J = 0 TO DY:F%(I,J) = 210: NEXT J,I
410  DATA 1,0,0,1,-1,0,0,-1
420  FOR I = 0 TO 3: READ XX%(I),YY%(I): NEXT
500 W(0) = 5:W(1) = 7:W(2) = 3:W(3) = 2
600  HGR2
700  HCOLOR= 3
800  FOR I = 0 TO (DX + 1) * 7 STEP 7
900  HPLOT I,0 TO I,(DY + 1) * 7
1000  NEXT
1100  FOR I = 0 TO (DY + 1) * 7 STEP 7
1200  HPLOT 0,I TO (DX + 1) * 7,I
1300  NEXT
1400 SP =  INT ( RND (1) * (DX + 1))
1500  HCOLOR= 0: HPLOT SP * 7 + 1,(DY + 1) * 7 TO SP * 7 + 6,(DY + 1) * 7
1600 F%(SP,DY) = F%(SP,DY) / 7
1700 SX = SP:SY = DY
1800 NZ = 0
1900 AX = SX:AY = SY
2000  GOSUB 9100
2100  HCOLOR= 3: HPLOT SX * 7 + 3,SY * 7 + 3: HPLOT SX * 7 + 4,SY * 7 + 4
2200  HCOLOR= 0:AX = SX:AY = SY
```



```
2300 WF = 0
2305  FOR D = 0 TO 3
2330 TX = SX + XX%(D):TY = SY + YY%(D)
2335  IF TX > DX OR TX < 0 THEN 2350
2340  IF TY > DY OR TY < 0 THEN 2350
2345  IF F%(TX,TY) = 210 THEN D%(WF) = D:WF = WF + 1
2350  NEXT
2400  IF WF = 0 THEN 5400
2450 ZZ = ZZ + 1
2500 D =  INT ( RND (1) * WF):D = D%(D)
3000 TX = SX + XX%(D):TY = SY + YY%(D)
3500 F%(SX,SY) = F%(SX,SY) / W(D)
3600  ON D + 1 GOSUB 9000,9010,9020,9030
4000 D = D + 2: IF D > 3 THEN D = D - 4
4100 F%(TX,TY) = F%(TX,TY) / W(D)
4200 SX = TX:SY = TY
4300 NX%(NZ) = SX:NY%(NZ) = SY:NZ = NZ + 1:HZ = NZ: GOTO 2000
5400  IF ZZ > DI * .9 THEN 5600
5500 Q =  RND (1) * NZ:SX = NX%(Q):SY = NY%(Q): GOTO 2000
5600 NZ = NZ - 1: IF NZ < 0 THEN 6000
5800 SX = NX%(NZ):SY = NY%(NZ): GOTO 2000
6000 SX = SP:SY = DY:AX = SX:AY = SY
6040  GOSUB 9100
7900  GET A$: RUN
9000  HPLOT (SX + 1) * 7,SY * 7 + 1 TO (SX + 1) * 7,SY * 7 + 6: RETURN
9010  HPLOT SX * 7 + 1,(SY + 1) * 7 TO SX * 7 + 6,(SY + 1) * 7: RETURN
9020  HPLOT SX * 7,SY * 7 + 1 TO SX * 7,SY * 7 + 6: RETURN
9030  HPLOT SX * 7 + 1,SY * 7 TO SX * 7 + 6,SY * 7: RETURN
9100  HCOLOR= 0: HPLOT AX * 7 + 3,AY * 7 + 3: HPLOT AX * 7 + 4,AY * 7 + 4
9110  HCOLOR= 3: HPLOT SX * 7 + 3,SY * 7 + 3: HPLOT SX * 7 + 4,SY * 7 + 4
9120  HCOLOR= 0:AX = SX:AY = SY: RETURN
```





Ausdruck des Apple-Bildschirms

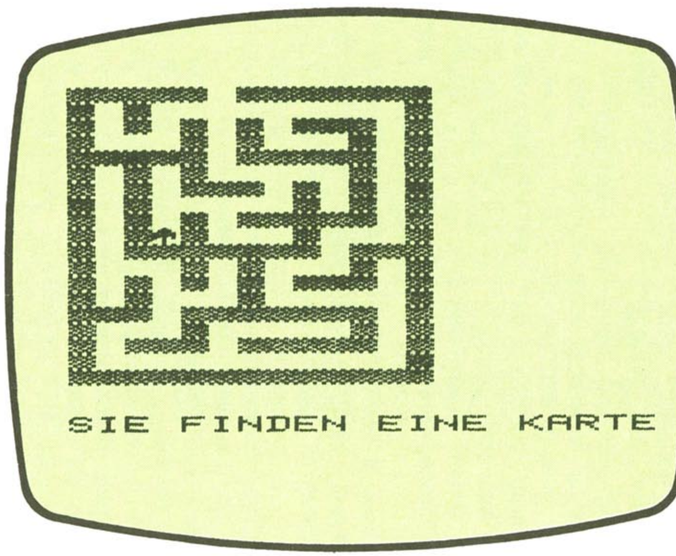
3-D-Labyrinth für VC-20

Im Heft 3/83 haben wir ein 3-D-Labyrinth für den VC-64 veröffentlicht. Eine große Anzahl von VC-20-Anwendern fragte uns, wie sie dieses Programm für ihren Rechner umschreiben könnten. Wir haben - weil es so schön in diese Rubrik passt - kurzerhand die nötigen Veränderungen vorgenommen und nun läuft dieses Programm auch auf dem VC-20.

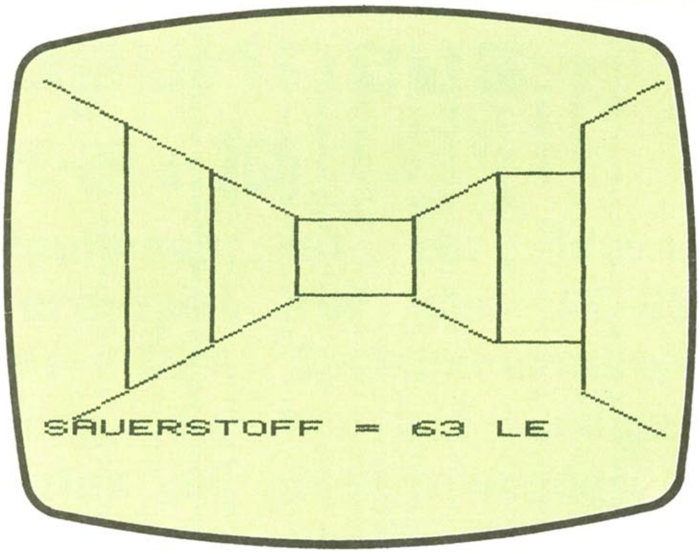
Der Spieler befindet sich in einem 3-D-Labyrinth durch das er sich mittels Druck auf die Tasten
<R> = nach rechts drehen
<L> = nach links drehen
<G> = geradeaus gehen
bewegen kann.
<M> = mogeln!
<P> = Print Bildschirm aus (Seikosha GP 100-VC)

Labyrinthe

```
3 POKE52,64:POKE54,64:POKE56,64
4 VP=33792:CO=5
5 GOTO900
10 Y=PEEK(TC-U):Z=PEEK(TC-U+LL):IFABS(U)=1ORDC=0THEN30
20 IFY=32ANDZ=32THENFORA=TC-U+LLTOBC-V-LLSTEPLL:POKEA,W3:NEXT
30 POKETC,TW:POKEBC,BW:POKETC+VP,CO:POKEBC+VP,CO
31 B=B+1:IFB<DWTHENTC=TC+U:BC=BC+V:GOTO30
40 IFABS(U)=1THENPOKETC,T1:POKEBC,B1:POKETC+VP,CO:POKEBC+VP,CO
50 IFTC=BC-LLTHENRETURN
60 FORB=TC+LLTOBC-LLSTEPLL:POKEB,SW:POKEB+VP,5:NEXT:RETURN
70 F=ML:U=LL+1:V=1-LL:T1=80:B1=122:SW=103
80 FORX=4TO6STEP2:TW=77:BW=78:B=0:IFX=4THEN100
90 F=MR:U=LL-1:V=-LL-1:SW=101:T1=79:B1=76:TW=78:BW=77
100 LM=M(F):LD=W(F)
101 IFF=3ANDOP=ECTHEN140
104 IFLD=0THENLD=2
105 IFQ-LD*INT(Q/LD)=0THENTC=M(X):BC=M(X+1):GOSUB10:GOTO130
110 TW=99:BW=100:U=U-LL:V=V+LL:N=PEEK(OP+LM)
120 IFN-D*INT(N/D)=0THENTC=M(X)+DW*LL:BC=M(X+1)+DW*-LL:GOSUB10
130 IFABS(U)=1THENU=U+LL:V=V-LL
140 M(X)=M(X)+DW*U:M(X+1)=M(X+1)+DW*V:NEXT
150 IFM=3ANDOP=ECTHENRETURN
160 IFQ-D*INT(Q/D)=0THEN180
170 OP=OP+M(M):Q=PEEK(OP):DC=DC+1:IFDC<CDTHEN70
175 IFDC=CDTHENRETURN
180 R=PEEK(M(4)-1)
185 IFR=32THENM(4)=M(4)-1:M(5)=M(5)-1:IFM(4)<>TL+DW*LLTHEN180
190 R=PEEK(M(6)+1)
195 IFR=32THENM(6)=M(6)+1:M(7)=M(7)+1:IFM(6)<>TR+DW*LLTHEN190
200 FORTC=M(4)TOM(6):POKETC,W1:POKETC+VP,CO:NEXT
210 FORBC=M(5)TOM(7):POKEBC,W2:POKEBC+VP,CO:NEXT:RETURN
240 PRINT"SAUERSTOFF ="OX"LE";
245 GETA$:IFA$=""THEN245
246 IFA$="P"THENGOSUB5000 :GOTO245
247 IFA$="M"THENGOSUB600:GOTO245
248 PRINT
250 NP=P:Q=PEEK(P):IFA$="G"THENNP=P+M(M)
260 IFNP=PTHEN320
270 IFNP=ECTHENPRINT"AUSGANG ERREICHT!":GOSUB4000:RUN
285 IFQ-D*INT(Q/D)><0THENP=NP:PRINT"SIE GEHEN":GOTO380
300 IFRND(1)>.9THENPRINT"IHR RAUMANZUG WIRD ZERSTOERT";:OL=OL+1
310 PRINT"SIE HABEN EINE WAND GERAMMT":GOSUB3000:GOTO410
320 NM=M:IFA$="R"THENNM=M+1
330 IFA$="O"THENNM=M+2
340 IFA$="L"THENNM=M+3
350 IFNM=MTHEN240
360 IFNM>3THENNM=NM-4*INT(NM/4)
370 M=NM:PRINT"SIE DREHEN SICH"
380 D=W(M):Q=PEEK(P):OP=P:ML=M-1:MR=M+1
390 IFML<0THENML=3
400 IFMR>3THENMR=0
410 OX=OX-OL:IFOX<0THENPRINT"IHRE LUFT IST VERBRAUCHT!":GOSUB4000:RUN
420 T=T+1:IFT=TTTHENTT=1:PRINT"SIE FINDEN EINE LAMPE":GOSUB3000
430 IFTT=1THENDC=0:GOSUB2000:GOSUB70
435 IFP=MCANDRM=0THENRM=1:GOSUB600
440 M(4)=TL:M(5)=BL:M(6)=TR:M(7)=BR:GOTO240
500 P=P+1:IFP>EMTHENP=SM
510 A=INT(RND(1)*4):DC=0
520 A=A+1:DC=DC+1:IFDC>3THEN500
530 IFA>3THENA=0
540 M=P+M(A):IFM<SMORM>EMTHEN520
550 CP=PEEK(P):CM=PEEK(M):IFC>0ANDCP=210THENP=M:GOTO510
560 TM=M-SM:IF(CP=CMORCM<210)ANDC>0THEN520
570 ME=TM-L*INT(TM/L):IF(ME=0ANDM(A)=1)OR(ME=GANDM(A)=-1)THEN520
580 OD=INT(15/W(A)):CP=CP/W(A):POKEP,CP:CM=CM/OD:POKEM,CM
590 P=M:C=C+1:IFC<HTHEN510
595 RETURN
600 GOSUB2000:PRINT"SIE FINDEN EINE KARTE"
605 A=TL+LL+1:MS=A:DC=SM:DD=SM+G
610 FORB=DCTODD:FORC=0TO3:CB=PEEK(B):CB=CB-W(C)*INT(CB/W(C))
620 AA=1:IFC=1ORC=3THENAA=LL
630 IFC>1THENAA=-AA
640 BB=LL/AA:P2=A+AA:P1=P2+BB:P3=P2-BB
645 IFB=PTHENPOKEA,W4:POKEA+VP,CO
650 IFCB=0THENPOKEP1,W3:POKEP2,W3:POKEP3,W3
651 IFCB=0THENPOKEP1+VP,CO:POKEP2+VP,CO:POKEP3+VP,CO
660 NEXT:A=A+2:NEXT:DC=DC+L:DD=DD+L:A=MS+(2*LL):MS=A
670 IFDD<=EMTHEN610
680 GOSUB3000:RETURN
900 GOSUB2000:PRINT"SIE VERSTECKEN SICH   AUF EINEM PLANETEN IN EINEM LABYRINTH"
910 PRINT"FINDEN SIE DEN AUGANG BEVOR IHNEN DER SAUER-STOFF AUSGEHT."
950 TL=4096:BL=4580:TR=TL+21:BR=BL+21:LL=22:DW=3:CD=3
955 W1=99:W2=100:W3=102:W4=88
960 W(0)=5:W(1)=7:W(2)=3:W(3)=2:L=INT(RND(1)*5+6)
970 M(0)=1:M(1)=L:M(2)=-1:M(3)=-L:M(4)=TL:M(5)=BL:M(6)=TR:M(7)=BR
980 W=INT(RND(1)*5+6):H=L*W-1:G=L-1:C=0:DC=0:T=0:RM=0
995 SM=16384:EM=SM+H:FORA=SMTOEM:POKEA,210:NEXT:P=SM+INT(RND(1)*H)
1000 EC=SM+INT(RND(1)*L):MC=INT(RND(1)*H)+SM
1010 OX=INT(H/3)+1:IFP<OXTHENOX=-OX
1015 OX=OX+H:OL=1:TT=INT(RND(1)*H/8)+1:GOSUB500:GOSUB2000
1030 M=0:NP=P:CE=PEEK(EC):CE=CE/W(3):POKEEC,CE:GOTO380
2000 PRINT"[illegible]"
3000 FORX=0TO5000:NEXT:RETURN
4000 GETA$:IFA$=""THEN4000
4010 RETURN
5000 OPEN4,4:PRINT#4,CHR$(14):BS$=CHR$(8):SI$=CHR$(14)
5005 FORI=4096TO4580STEP22:P$=SI$
5010 FORJ=0TO21:X=PEEK(J+I)
5020 IFX<32THENX=X+64:GOTO5100
5025 IFX<64THEN5100
5030 IFX<96THENX=X+32:GOTO5100
5035 IFX<128THENX=X+64:GOTO5100
5100 P$=P$+CHR$(X)
5150 NEXT:PRINT#4,P$;BS$:NEXT
5155 CLOSE4
5160 RETURN
```



Ausdruck des VC-20-Bildschirms



Ausdruck des VC-20-Bildschirms

Maze-Challenger

Um die Labyrinthrubrik für diese Ausgabe abzuschließen noch ein Programm für den CBM 3032. Es erklärt sich nach dem Starten von selbst, was uns auf eine genaue Spielerklärung verzichten ließ.
Die Idee kann - weil das eigentliche Spielunterprogramm recht kurz ist - leicht für andere Rechner umgeschrieben werden.

```
1 poke59468,14
2 poke59490,30
5 print"▓"
10 fori=32768to32808:gosub70:next
20 i=i+38:gosub70:i=i+1:gosub70
30 print:print"▓## ▓ Maze-Challenger
                    ▓## ▓"
40 print"▓▓## ▓
              ▓## ▓▓▓▓## ▓"

50 fori=32968to33007:gosub70:next
55 print:print"                                    H
omecomputer ":goto71
70 pokei,163:return
71 print"▓So wird's gemacht:"
72 print"Mittels der - den Bewegungsrich
tungen"
73 print"entsprechenden Tasten:"
74 print" ▓ 4 ▓ ▓ 6 ▓ ▓ 8 ▓ & ▓ 2 ▓"
75 print"kann ein Kaefer durch ein Labyr
inth di- rigiert werden."
76 print"Wer's am schnellsten kann ist S
ieger!"
77 print"Am Ende wird nicht nur die beno
etigte"
78 print"Zeit bewertet, sondern entschei
dend ist,";
79 print"wer die meisten 'Spruenge pro S
tunde'!"
80 print"(JP/H) schaffte."
81 print"▓▓Druecke ▓ ↑ ▓"
90 geta$:ifa$=""then90
91 ifa$<>"↑"then90
92 print"▓ ▓▓▓▓ich berechne das labyri
nth!":poke59468,12
100 h=19:v=10
110 dimw(h,v),v(h,v)
130 print"▓":print:print
140 poke59490,255
160 q=0:z=0:x=int(rnd(1)*h+1)
165 fori=1toh
170 ifi=xthen173
171 print"##";:goto180
173 print"# ";
180 next
190 print"#"
195 c=1:w(x,1)=c:c=c+1
200 r=x:s=1:goto260
210 ifr<>hthen240
215 ifs<>vthen230
220 r=1:s=1:goto250
230 r=1:s=s+1:goto250
240 r=r+1
```

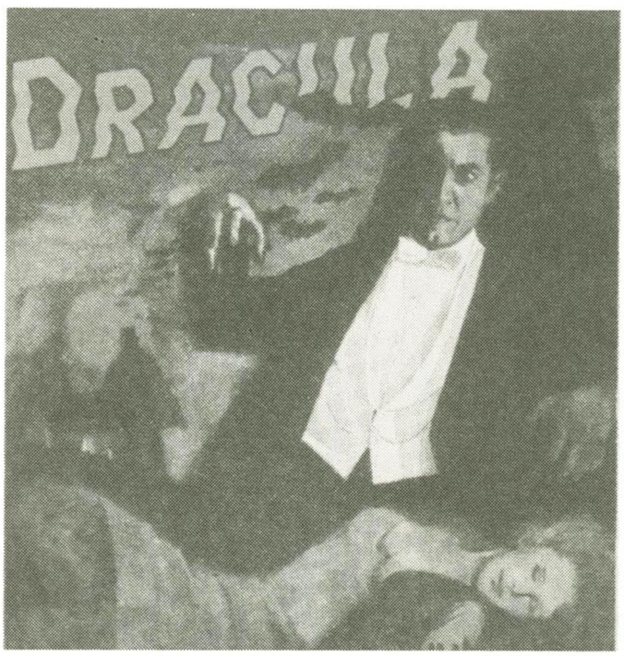
Labyrinthe

```
250 ifw(r,s)=0then210
260 ifr-1=0then530
265 ifw(r-1,s)<>0then530
270 ifs-1=0then390
280 ifw(r,s-1)<>0then390
290 ifr=hthen330
300 ifw(r+1,s)<>0then330
310 x=int(rnd(1)*3+1)
320 onxgoto790,820,860
330 ifs<>vthen340
334 ifz=1then370
338 q=1:goto350
340 ifw(r,s+1)<>0then370
350 x=int(rnd(1)*3+1)
360 onxgoto790,820,910
370 x=int(rnd(1)*2+1)
380 onxgoto790,820
390 ifr=hthen470
400 ifw(r+1,s)<>0then470
405 ifs<>vthen420
410 ifz=1then450
415 q=1:goto430
420 ifw(r,s+1)<>0then450
430 x=int(rnd(1)*3+1)
440 onxgoto790,860,910
450 x=int(rnd(1)*2+1)
460 onxgoto790,860
470 ifs><vthen490
480 ifz=1then520
485 q=1:goto500
490 ifw(r,s+1)<>0then520
500 x=int(rnd(1)*2+1)
510 onxgoto790,910
520 goto790
530 ifs-1=0then670
540 ifw(r,s-1)<>0then670
545 ifr=hthen610
547 ifw(r+1,s)<>0then610
550 ifs><vthen560
552 ifz=1then590
554 q=1:goto570
560 ifw(r,s+1)<>0then590
570 x=int(rnd(1)*3+1)
580 onxgoto820,860,910
590 x=int(rnd(1)*2+1)
600 onxgoto820,860
610 ifs><vthen630
620 ifz=1then660
625 q=1:goto640
630 ifw(r,s+1)<>0then660
640 x=int(rnd(1)*2+1)
650 onxgoto820,910
660 goto820
670 ifr=hthen740
680 ifw(r+1,s)<>0then740
685 ifs><vthen700
690 ifz=1then730
695 q=1:goto830
700 ifw(r,s+1)<>0then730
710 x=int(rnd(1)*2+1)
720 onxgoto860,910
730 goto860
740 ifs<>vthen760
750 ifz=1then780
755 q=1:goto770
760 ifw(r,s+1)<>0then780
770 goto910
780 goto1000
790 w(r-1,s)=c
800 c=c+1:v(r-1,s)=2:r=r-1
810 ifc=h*v+1then1010
815 q=0:goto260
820 w(r,s-1)=c
830 c=c+1
840 v(r,s-1)=1:s=s-1:ifc=h*v+1then1010
850 q=0:goto260
860 w(r+1,s)=c
870 c=c+1:ifv(r,s)=0then880
875 v(r,s)=3:goto890
880 v(r,s)=2
890 r=r+1
900 ifc=h*v+1then1010
905 goto530
910 ifq=1then960
920 w(r,s+1)=c:c=c+1:ifv(r,s)=0then940
930 v(r,s)=3:goto950
940 v(r,s)=1
950 s=s+1:ifc=h*v+1then1010
955 goto260
960 z=1
970 ifv(r,s)=0then980
975 v(r,s)=3:q=0:goto1000
980 v(r,s)=1:q=0:r=1:s=1:goto250
1000 goto210
1010 forj=1tov
1011 print"#";
1012 fori=1toh
1013 ifv(i,j)<2then1030
1020 print"  ";
1021 goto1040
1030 print" #";
1040 nexti
1041 print
1043 fori=1toh
1045 ifv(i,j)=0then1060
1050 ifv(i,j)=2then1060
1051 print"# ";
1052 goto1070
1060 print"##";
1070 nexti
1071 print"#"
1072 nextj
1080 r=2:s=0
1100 poke32848,170
1110 ti$="000000":t1=ti
1120 print"█";ti$:geta$:ifa$="2"thenr=r+
1:f1=-1:f2=0:goto1160
1130 ifa$="4"thens=s-1:f1=0:f2=1:goto116
0
1140 ifa$="6"thens=s+1:f1=0:f2=-1:goto11
60
1150 ifa$="8"thenr=r-1:f1=1:f2=0:goto116
0
1155 goto1120
1160 ifpeek(32768+r*40+s)=35thenr=r+f1:s
=s+f2:gosub2000
1165 ifpeek(32768+r*40+s)=170thenpoke(32
768+(r+f1)*40+(s+f2)),160
1170 poke(32768+r*40+s),170
1180 ifr=24thentc$=ti$:t1=ti-t1:goto1200
1190 goto1120
1200 print"█ready after ";tc$;" hhmmss."
1202 c=0
1204 fori=32848to33767:ifpeek(i)=170orpe
ek(i)=42thenc=c+1
1206 next
1208 print"speed was";int(c/t1*216000);"
jp/h (jumps/hour)"
1209 print"[graphics]
 cmd: 'n'"
1210 geta$:ifa$><"n"then1210
1220 run
2000 t=ti
2010 ifti-t<120thenpoke(32768+r*40+s),16
0:goto2010
2020 return
```

Spukhaus

VC-20 + 16K

Spukhaus ist ein Adventure-Spiel für den VC-20 mit 16K Speichererweiterung, die bis auf wenige Bytes voll genutzt werden. Sinn des Spieles ist es, Jane, die von Dracula und seinem Gehilfen entführt und im Spukhaus versteckt worden ist, wieder zu befreien.



Insgesamt sind 7 verschiedene Räume in veränderlicher Anordnung vorhanden:

Eingangshalle
Esszimmer
Schlafzimmer
Speisekammer
Dachboden
Labor
Gruft

Die Räume sind graphisch mit Hilfe von selbstdefinierten Zeichen (unter Verwendung des Zeichengenerators aus Homecomputer 3/83) gestaltet worden.

Wichtig: Vor dem LOAD-Befehl muß daher mit POKE 43,227: POKE44,19: POKE5090,0 der Startzeiger auf den veränderten Basicstart eingestellt werden.

Wichtige Spielpassagen werden akustisch angezeigt. Da der Spieler zudem gegen die Zeit spielt, kann Langeweile nicht auftreten.

Das Spiel ist so aufgebaut, daß Befehlswörter und Sätze in einfachem Deutsch akzeptiert werden.
Mögliche Befehlswörter sind:
Für die Richtungen Norden, Süden, Westen und Osten jeweils der Anfangsbuchstabe: N, S, W und O
GEH TUER
GEH TREPPE
GEH LOCH
GEH LUKE
GEH BALKON
HOCH
SIEH FENSTER
SIEH TISCH
SIEH SCHRANK
SIEH KAMIN
SIEH STUHL
SIEH TEPPICH
SIEH WAND
SIEH BILD
SIEH REGAL
SIEH TRUHE
NIMM STREICHHOLZ
NIMM KREUZ
NIMM HAMMER
NIMM PFLOCK
NIMM KERZE
NIMM MESSER
NIMM SCHLUESSEL
ISS
TRINK
SCHLAF
LIES ZETTEL
OEFFNE TUER
OEFFNE LUKE
ZIEH HEBEL
DRUECK HEBEL
MIT SCHLUESSEL
MIT HAMMER
MIT PFLOCK
KILL DRACULA
KILL GEHILFE

Mittels der Befehlwörter kann man von einem Zimmer in ein anderes gelangen oder Anweisungen geben (essen, trinken, schlafen).

Bei Anordnungen des Computers wie DU MUSST ESSEN hat man 2 Spielstunden (= 2 Minuten Echtzeit) Zeit für die Ausführung des Befehls.
Das Spiel beginnt um 14.00 Uhr Spielzeit. Um 18.00 beginnt die Dämmerung. Wenn der Spieler bis 20 Uhr nicht Streichhölzer und Kerze gefunden hat, wird es (auf dem Bildschirm) ganz dunkel, was das Spiel natürlich enorm erschwert.
Hat man schließlich, nachdem man mindestens einmal in der Gruft war, Pflock und Hammer gefunden, ist es möglich, Dracula auf echte Vampirart zu töten; natürlich nur am Tage zwischen 8 und 17.00 Uhr (nachts ist er ja nicht im Sarg).
Erscheint Dracula nachts (zwischen 0 und 6 Uhr) und hat der Spieler nicht innerhalb einer Stunde das Kreuz zur Abwehr bereit, um es Dracula zu zeigen, so wird man von ihm gebissen und hat verloren. Arme Jane ...
Weiß man einmal nicht so recht weiter oder will das Spiel mal kurz unterbrechen (z.B. weil die Schularbeiten noch nicht gemacht sind) kann man dies mit dem Befehl WARTE tun. Das Spiel bleibt unterbrochen, bis SPACE gedrückt wurde.
Übrigens, wenn sich malnach einem Befehl nichts tut, liegt es vielleicht daran, daß kein RETURN gedrückt wurde (der Cursor ist nämlich im neuen Zeichensatz verschwunden).
Alles weitere ergibt sich beim Spielen. Sollten Sie beim ersten Mal ohne Jane und mit Bißwunden am Hals enden, nicht verzweifeln. Einer meiner Freunde hat es schon einmal geschafft!
Spaß und Horror sind garantiert!!!

Gregor Widuch

```
5 DIMZ$(7,13)
10 PRINT"□"
20 POKE36869,205:POKE36879,127:LI=0:POKE650,0
30 GOSUB50000
35 YY=INT(RND(1)*3)+1
40 IFYY=1THENLT=1
42 IFYY=2THENZT=1
44 IFYY=3THENET=1
50 TI$="000000":ZE=14:ZN=ZE+6:GZ=30
990 GOTO5000
1000 ZI=1:GOSUB45000
1010 IFSE=1THENPRINTQ$;"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"
1020 IFDB=1THENPRINTQ$;"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒!"
1030 GOSUB49000
1040 PRINTQ$"DU BIST IN DER"
1045 PRINT"EINGANGSHALLE"
1047 FORQ=1TO2000:NEXT:GOSUB49000
1050 GOSUB21100
1055 GOSUB49000:GOSUB14000
1060 IFVN$="N"ORVN$="GEH TUER"THENONYYGOTO3000,2000,7000
1065 IFVN$="S"THENGOSUB40100:H=1
1070 IFVN$="O"THENGOSUB40000:H=1
1075 IFVN$="W"ORVN$="GEH TREPPE"ORVN$="HOCH"THENONYYGOTO8000,4000,6000
1077 IFLG-6=<0THEN1085
1080 IFLEFT$(VN$,4)="SIEH"ANDMID$(VN$,LG-6,7)="FENSTER"THENGOSUB41000:H=1
1085 IFSE=1ANDVN$="NIMM STREICHHOLZ"THENSE=0:SH=2:GOSUB40700:H=1
1090 IFVN$=""THEN1050
1100 REM
1120 IFDB=1ANDVN$="NIMM KREUZ"THENDB=0:H=1:GOSUB40700
1890 IFH=1THENFORQ=1TO2000:NEXT:H=0:GOTO1050
1895 GOSUB40200
1900 FORQ=1TO2000:NEXT:GOTO1050
2000 ZI=4:GOSUB45000
2010 IFAA=1THENPRINTQ$;"▒▒▒▒▒▒▒▒▒▒▒▒▒▒"
2040 GOSUB49000
2050 PRINTQ$;"DU BIST IN DER"
2055 PRINT"SPEISEKAMMER"
2060 FORQ=1TO2000:NEXT:GOSUB49000
2100 GOSUB21100
2105 GOSUB49000:GOSUB14000
2110 IFVN$="S"THENONYYGOTO3000,1000,6000
2120 IFVN$="GEH LOCH"ORVN$="N"THENONYYGOTO10000,7000,10000
2125 IFLG-6=<0THEN2140
2130 IFHB<>1ANDLEFT$(VN$,4)="SIEH"ANDMID$(VN$,LG-6,7)="SCHRANK"THENGOSUB41000:H=
1
2135 IFHB=1ANDLEFT$(VN$,4)="SIEH"ANDMID$(VN$,LG-6,7)="SCHRANK"THENGOSUB21000
2140 IFVN$=""THEN2100
2150 IFVN$="W"ORVN$="O"THENGOSUB40000:H=1
2160 IFHB=1ANDVN$="NIMM HAMMER"THENH=1:HB=2:GOSUB40700
2170 IFES=1ANDVN$="ISS"THENPRINTQ$;"OK HAST GEGESSEN":ES=0:H=1
2180 IFTR=1ANDVN$="TRINK"THENPRINTQ$;"OK HAST GETRUNKEN":TR=0:H=1
2200 IFAA=1ANDVN$="LIES ZETTEL"THENH=1:AA=0:GOSUB41500:H=1:SS=1
2900 IFH=1THENFORQ=1TO2000:NEXT:H=0:GOTO2100
2910 GOSUB40200:FORQ=1TO2000:NEXT:GOTO2100
3000 ZI=2:GOSUB45000
3010 IFKS=1THENPRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒ "
3020 IFAB=1THENPRINTQ$;"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"
3040 GOSUB49000
3050 PRINTQ$;"DU BIST IM ESSZIMMER"
3060 FORQ=1TO2000:NEXT:GOSUB49000
3070 H=0:GT=0:LZ=0
3100 GOSUB21100
3105 GOSUB49000:GOSUB12000:GOSUB14000
3110 IFVN$="S"THENONYYGOTO1000,4000,4000
3120 IFVN$="O"ORVN$="N"ANDLZ=0ORVN$="W"ANDGT=0THENGOSUB40000:H=1
3125 IFLG-4=<0THEN3150
3130 IFLEFT$(VN$,4)="SIEH"ANDMID$(VN$,LG-4,5)="TISCH"THENGOSUB21010
3150 IFVN$="GEH LUKE"ANDLZ=0THENPRINTQ$;"IST VERSCHLOSSEN":H=1
3170 IFVN$="OEFFNE LUKE"THENGOSUB40600:LZ=1:H=1
3175 IFLG-4=<0THEN3200
3180 IFLEFT$(VN$,4)="SIEH"ANDMID$(VN$,LG-4,5)="KAMIN"THENGOSUB21020
3200 IFVN$="ZIEH HEBEL"THENPRINTQ$;"GEHT NICHT":H=1
3210 IFVN$="DRUECK HEBEL"THENGOSUB40300:GT=1:H=1
3230 IFLZ=1THENIFVN$="GEH LUKE"ORVN$="N"THENONYYGOTO2000,10000,10000
3240 IFVN$=""THEN3100
3250 IFGT=1ANDET=0THENIFVN$="GEH TUER"ORVN$="W"THENONYYGOTO6000,8000,41100
3260 IFGE=1ANDET=1THENIFVN$="GEH TUER"ORVN$="W"THEN13000
3270 IFGE=1ANDET=1THENIFVN$="OEFFNE TUER"THENGOSUB40500:H=1
3280 IFGT=1THENIFET=1THENIFVN$="OEFFNE TUER"THENGOSUB40500:H=1
3290 IFGT=1THENIFET=1THENIFVN$="GEH TUER"ORVN$="W"THENGOSUB40400:H=1
3295 IFGT=1ANDSC=1AND(VN$="W"ORVN$="GEH TUER")THEN13000
3300 IFKS=1ANDVN$="NIMM KERZE"THENGOSUB40700:KS=0:H=1:KE=2
3310 IFAB=1ANDVN$="LIES ZETTEL"THENH=1:GOSUB41500:AB=0
3320 IFLG-4=<0THEN3420
3400 IFBB<>1ANDLEFT$(VN$,4)="SIEH"ANDMID$(VN$,LEN(VN$)-4,5)="STUHL"THENGOSUB4100
0:H=1
3410 IFBB=1ANDLEFT$(VN$,4)="SIEH"ANDMID$(VN$,LEN(VN$)-4,5)="STUHL"THENGOSUB21030
3420 IFBB=1ANDVN$="NIMM SCHLUESSEL"THENH=1:SC=1:BB=0:GOSUB40700
3800 IFH=1THENFORQ=1TO2000:NEXT:H=0:GOTO3100
3900 GOSUB40200:FORQ=1TO2000:NEXT:GOTO3100
4000 ZI=3:GOSUB45000
4010 IFKL=1THENPRINTQ$;"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒ "
4030 GOSUB49000
4040 PRINTQ$;"DU BIST IM LABOR"
4047 FORQ=1TO2000:NEXT:GOSUB49000
4100 GOSUB21100
4105 GOSUB49000:GOSUB12000:GOSUB14000
4110 IFGE=1ANDLT=1THENIFVN$="N"ORVN$="GEH TUER"THEN13000
4112 IFGE=1ANDLT=1THENIFVN$="OEFFNE TUER"THENGOSUB40500:H=1
4120 IFVN$=""THEN4100
4125 IFLT=1THENIFVN$="OEFFNE TUER"THENGOSUB40500:H=1
4127 IFLT=1THENIFVN$="GEH TUER"ORVN$="N"THENGOSUB40400:H=1
4130 IFLT=0THENIFVN$="OEFFNE TUER"ORVN$="N"THENONYYGOTO41100,3000,3000
4140 IFVN$="W"ORVN$="S"THENGOSUB40200:H=1
4150 IFVN$="O"THENONYYGOTO10000,1000,7000
4160 IFKL=1ANDVN$="NIMM KERZE"THENGOSUB40700:KE=2:H=1:KL=0
4190 IFLG-4=<0THEN4220
4200 IFLEFT$(VN$,4)="SIEH"ANDMID$(VN$,LG-4,5)="REGAL"ANDHA<>1THENGOSUB41000:H=1
4210 IFLEFT$(VN$,4)="SIEH"ANDMID$(VN$,LG-4,5)="REGAL"ANDHA=1THENGOSUB21000
4220 IFVN$="NIMM HAMMER"ANDHA=1THENGOSUB40700:H=1:HA=2
4230 IFVN$="TRINK"THENPRINT"▒▒▒DAS WAR GIFT":PRINT"▒DU BIST TOT":GOSUB20400:END
4900 IFH=1THENFORQ=1TO2000:NEXT:H=0:GOTO4100
4910 GOSUB40200:FORQ=1TO2000:NEXT:GOTO4100
5000 PRINT"▒▒▒▒DU STEHST VOR DEM      ▒▒▒SPUKHAUS":VN$=""
5010 PRINT"▒▒▒ES IST 14.00 UHR":PRINT"▒SUCHE JANE"
5030 FORQ=1TO10:POKE36878,15:POKE36874,200+Q*2:FORQQ=1TO100:NEXTQQ
5040 POKE36874,210+Q*2:FORQQ=1TO100:NEXTQQ,Q:FORQ=1TO23
5050 POKE36874,230+Q*1.1:POKE36875,230-Q*1.1:NEXTQ:POKE36874,0:POKE36875,0
5060 GOTO1000
6000 ZI=5:GOSUB45000
6010 IFSZ=1THENPRINTQ$;"▒▒▒▒▒▒▒▒▒▒▒▒"
6040 GOSUB49000
6045 IFGJ=1THENPRINTQ$;"▒▒▒▒▒▒▒▒!"
6050 PRINTQ$;"DU BIST IM SCHLAF-"
6055 PRINT"ZIMMER"
6060 FORQ=1TO2000:NEXT:GOSUB49000:H=0
6100 GOSUB21100
6105 GOSUB49000:GOSUB12000:GOSUB14000
```

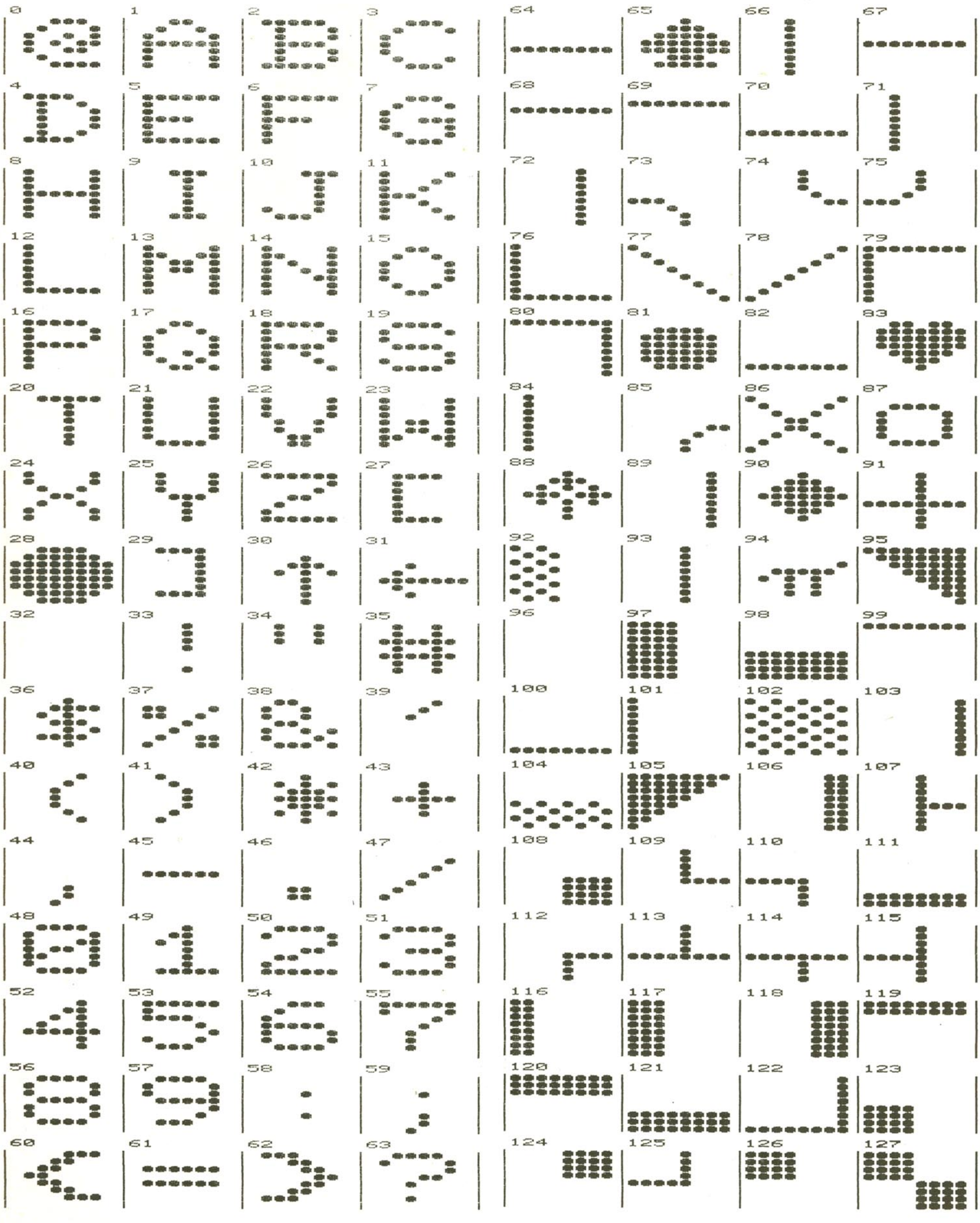
```
6107 IFGE=1THENIFZT=1THENIFVN$="OEFFNE TUER"THENGOSUB40500:H=1
6108 IFGE=1THENIFZT=1THENIFVN$="GEH TUER"ORVN$="N"THEN13000
6110 IFZT=0THENIFVN$="N"ORVN$="GEH TUER"THENONYYGOTO7000,41100,2000
6115 IFZT=1THENIFVN$="OEFFNE TUER"THENGOSUB40500:H=1
6117 IFZT=1THENIFVN$="GEH TUER"ORVN$="N"THENGOSUB40400:H=1
6120 IFVN$="O"ORVN$="GEH LOCH"THENONYYGOTO3000,7000,1000
6130 IFVN$="W"ORVN$="GEH BALKON"THEN9000
6135 IFLG-6=<0THEN6147
6140 IFLEFT$(VN$,4)="SIEH"ANDMID$(VN$,LG-6,7)="TEPPICH"ANDPF<>1THENGOSUB41000:H=
1
6145 IFLEFT$(VN$,4)="SIEH"ANDMID$(VN$,LG-6,7)="TEPPICH"ANDPF=1THENGOSUB21040
6147 IFVN$="NIMM PFLOCK"ANDPF=1THENGOSUB40700:H=1:PF=2
6149 IFLG-3=<0THEN6160
6150 IFLEFT$(VN$,4)="SIEH"ANDMID$(VN$,LG-3,4)="WAND"THENGOSUB41000:H=1
6155 IFLEFT$(VN$,4)="SIEH"ANDMID$(VN$,LG-3,4)="BILD"THENGOSUB41000:H=1
6160 IFVN$=""THEN6100
6170 IFPE=1ANDVN$="SCHLAF"THENGOSUB47000:PE=0:H=1
6180 IFGJ=1ANDVN$="NIMM MESSER"THENGJ=0:ME=0:H=1:PRINTQ$"OK HAST GENOMMEN"
6200 IFSZ=1ANDVN$="NIMM STREICHHOLZ"THENSZ=0:SH=2:GOSUB40700
6900 IFH=1THENFORQ=1TO2000:NEXT:H=0:GOTO6100
6910 GOSUB40200:FORQ=1TO2000:NEXT:GOTO6100
7000 GOSUB21200
7005 PRINT"DER WEG VERZWEIGT"
7010 PRINT"DU KANNST NACH         OSTEN  (O)              WESTEN (W)"
7020 PRINT"SUEDEN (S)
7030 K1=1:H=0:GOSUB21100
7035 GOSUB49000:GOSUB14000
7040 IFVN$="S"THENONYYGOTO6000,2000,1000
7050 IFVN$="O"THENONYYGOTO8000,10000,8000
7060 IFVN$="W"THENONYYGOTO10000,6000,4000
7070 IFVN$="N"THENGOSUB40000:H=1
7080 IFVN$=""THEN7030
7090 IFH=1THENFORQ=1TO2000:NEXT:H=0:GOTO7030
7100 GOSUB40200:FORQ=1TO2000:NEXT:GOTO7030
8000 ZI=6:GOSUB45000
8020 IFGI=1THENPRINT"!"
8025 IFDA=1THENPRINTQ$;"!"
8030 GOSUB49000
8040 PRINTQ$;"DU BIST AUF DEM"
8045 PRINT"DACHBODEN"
8047 FORQ=1TO2000:NEXT:GOSUB49000
8100 GOSUB21100
8105 GOSUB49000:GOSUB14000
8110 IFVN$="O"ORVN$="GEH LUKE"THENONYYGOTO1000,3000,10000
8115 IFVN$="W"ORVN$="GEH TUER"THENONYYGOTO7000,10000,7000
8120 IFVN$="N"ORVN$="S"THENGOSUB40000:H=1
8130 IFLG-4=<0THEN8144
8140 IFLEFT$(VN$,4)="SIEH"ANDMID$(VN$,LG-4,5)="TRUHE"THENIFPH<>1THENGOSUB41000:H
=1
8142 IFLEFT$(VN$,4)="SIEH"ANDMID$(VN$,LG-4,5)="TRUHE"THENIFPH=1THENGOSUB21040
8144 IFVN$="NIMM PFLOCK"ANDPH=1THENGOSUB40700:H=1:PH=2
8147 IFLG-4=<0THEN8157
8150 IFLEFT$(VN$,4)="SIEH"ANDMID$(VN$,LG-4,5)="PAKET"ANDB1=0THENGOSUB41000:H=1
8155 IFLEFT$(VN$,4)="SIEH"ANDMID$(VN$,LG-4,5)="PAKET"ANDB1=1THENGOSUB21030
8157 IFVN$="NIMM SCHLUESSEL"ANDB1=1THENB1=0:PRINTQ$"OK HAST GENOMMEN":H=1:SC=1
8160 IFVN$=""THEN8100
8180 IFGI=1ANDVN$="NIMM MESSER"THENME=0:GI=0:H=1:GOSUB40700
8200 IFDA=1ANDVN$="NIMM KREUZ"THENDA=0:H=1:GOSUB40700
8900 IFH=1THENFORQ=1TO2000:NEXT:H=0:GOTO8100
8910 GOSUB40200:FORQ=1TO2000:NEXT:GOTO8100
9000 PRINT"DU BIST IM GARTEN      DU KANNST NUR DURCH    DIE EINGANGSTUER "
9010 PRINT"WIEDER INS              SPUKSCHLOSS"
9020 FORQ=1TO4000:NEXT:GOTO1000
10000 GOSUB21200
10005 PRINT"DU BIST AUF DEM       FLUR                    DU KANNST NACH"
10010 PRINT"OSTEN (O)               SUEDEN (S),            WESTEN (W)"
10015 K1=1:H=0:GOSUB21100
10017 GOSUB49000:GOSUB14000
10020 IFVN$="O"THENONYYGOTO7000,8000,3000
10030 IFVN$="S"THENONYYGOTO2000,3000,2000
10040 IFVN$="W"THENONYYGOTO4000,7000,8000
10050 IFVN$="N"THENGOSUB40000:H=1
10060 IFH=1THENFORQ=1TO2000:NEXT:H=0:GOTO10015
10065 IFVN$=""THEN10015
10070 GOSUB40200:FORQ=1TO2000:NEXT:H=0
10099 GOTO10015
12000 REM
12050 IFVN$="MIT SCHLUESSEL"THENIFSC=1THENGE=1:H=1:BB=0:AA=0:GOSUB40600
12055 IFVN$="MIT SCHLUESSEL"THENIFSC=0THENPRINTQ$;"HAST KEINEN":H=1
12900 RETURN
13000 ZI=7:GOSUB45000
13040 GOSUB49000
13050 PRINTQ$;"DU BIST IN DER GRUFT"
13060 FORQ=1TO2000:NEXT:GOSUB49000
13070 H=0
13100 GOSUB21100:GOSUB49000
13105 GOSUB14000
13120 IFVN$="S"THENONYYGOTO4000,6000,3000
13130 IFHA=2ANDPF=2ORHA=2ANDPH=2ORHB=2ANDPF=2ORHB=2ANDPH=2THEN13140
13135 GOTO13150
13140 IFZE=>18ORZE=<7THEN13200
13145 IFVN$="KILL DRACULA"THENH=1:D3=1:GOTO13160
13150 IFD3=0THENGOTO13200
13160 IFDU=0THENPRINTQ$"WOMIT ?":H=1
13170 IFD1=0ANDVN$="MIT HAMMER"THENDU=DU+1:H=1:D1=1
13175 IFD2=0ANDVN$="MIT PFLOCK"THENDU=DU+1:D2=1:H=1
13177 IFDU=1ANDD4=0THENPRINTQ$"UND WOMIT NOCH ?":H=1:D4=1
13180 IFDU=2THEN13190
13185 GOTO13200
13190 PRINTQ$;"DU HAST DRACULA ":PRINT"GETOETET":DR=1:H=1:GOSUB20200
13200 REM
13900 IFH=1THENFORQ=1TO2000:NEXT:H=0:GOTO13100
13910 GOSUB40200:FORQ=1TO2000:NEXT:GOTO13100
14000 PRINTQ$"":LG=LEN(VN$):SL=0:II=INT(VAL(TI$)/100):IFII>0THEN14100
14050 GOTO14240
14100 FORIJ=1TOII:SL=1
14110 ZE=ZE+1:IFZE=>24THENZE=0
14120 ZN=ZN+1:IFZN=>24THENZN=0
14125 IFZN=2THENOP=OP+1:IFOP=1THENLI=0
14127 IFZN=3THENOP=0
14130 IFZN=>0ANDZN<2ORZN=>12ANDZN<14THENLI=1:POKE36879,8
14135 IFZN=>2ANDZN<12THENPOKE36879,8
14140 IFZN=14THENLI=0:POKE36879,127
14150 REM
14200 IFGH=1THEN14240
14202 IFGG=1THEN14300
14205 X=INT(RND(1)*5)+1:IFX=5THEN14210
14207 GOTO14240
14210 GOSUB20100:GH=1:X=INT(RND(1)*2)+1:IFX=1THENGI=1
14215 IFX=2THENGJ=1
14220 ME=1:GZ=ZE:GK=ZI:PRINT"SUCH DAS MESSER":HH=1
14240 REM
14250 IFGZ=ZEANDME=0ANDGK=ZIANDVN$="KILL GEHILFE"THEN14255
14252 GOTO14260
14255 X=INT(RND(1)*2)+1:IFX=1THEN14258
14256 PRINT"DER GEHILFE IST VER-":PRINT"JAGT":GOSUB20300
14257 HH=1:H=1:GH=0:GK=0:GZ=30:GOSUB45001:GOTO14260
14258 PRINT"DER GEHILFE IST TOT":HH=1:GZ=30:H=1:GG=1:GK=0:GOSUB20200:GOSUB45001
```

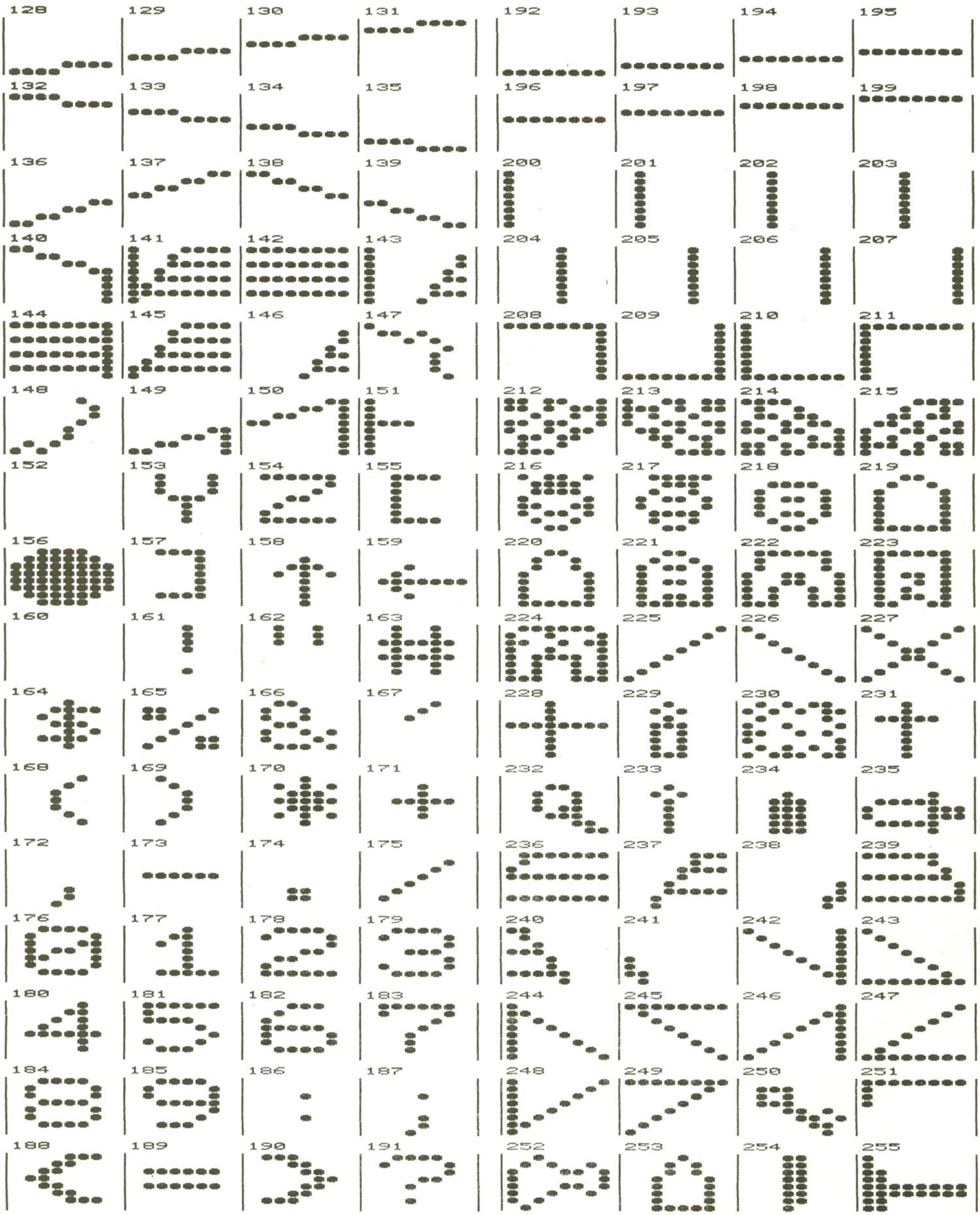
```
14260 IFGZ+1=ZEANDME=0THENGH=0:HH=1:GK=0:GZ=30:GOTO14267
14265 GOTO14270
14267 PRINT"DU HAST DEN GEHILFEN":PRINT"VERJAGT":GOSUB20300:GOSUB45001
14270 IFGZ+2=ZEANDME=1THEN46300
14300 IFGG=1THEN14400
14305 IFGK<>ZIORK1=1THEN14340
14310 PRINTQ$"[illegible]";CHR$(17);"[illegible]";CHR$(17);"[illegible]":PRINTQ$"[illegible]"
14340 REM
14400 IFZE=DE+1THENDQ=0
14401 IFII<1THEN14430
14402 IFZE=>6ORDR=1THEN14500
14405 IFDD=1THEN14430
14407 X=INT(RND(1)*3)+1:IFX<>2THEN14500
14408 DQ=DQ+1:IFDQ=1THENPRINT"SUCH DAS KREUZ":HH=1:GOSUB20100
14410 DD=1:DZ=ZI:DE=ZE:X=INT(RND(1)*2)+1:IFX=1THENDA=1:GOTO14430
14420 DB=1
14430 IFZE=DEANDDZ=ZITHENIFDA=0ANDDB=0THENIFVN$="ZEIG KREUZ"THEN14440
14432 IFZE=DE+1THENIFDD=1THEN46400
14435 GOTO14450
14440 PRINT"DU HAST DRACULA":PRINT"VERJAGT":HH=1:DD=0:DZ=0:H=1:GOSUB20300:GOSUB4
5000
14450 IFDZ<>ZIORK1=1ORDD=0ORZE>6THEN14500
14460 PRINTQ$;"[illegible]";CHR$(17);"[illegible]";CHR$(17);"[illegible]":PRINTQ$"[illegible]
[illegible]"
14500 REM
15000 GOSUB21200
15005 IFZE=DE+1THENDQ=0
15010 IFZL=1THENSU=SU+1:IFSU=1THENPRINT"SUCH DEN ZETTEL":GOTO15015
15012 GOTO15020
15015 X=INT(RND(1)*2)+1:IFX=1THENAA=1:GOTO15017
15016 AB=1
15017 HH=1
15020 IFZE<21ANDSS=1ANDAD=ZETHENSQ=SQ+1:IFSQ=1THENHH=1:GOTO15030
15025 IFZE=>21ANDSS=1THENIFAD=>20THENSQ=SQ+1:IFSQ=1THENHH=1:GOTO15030
15027 GOTO15100
15030 PRINT"SUCH DEN SCHLUES-":PRINT"SEL JETZT"
15040 X=INT(RND(1)*2)+1:IFX=1THENBB=1:GOTO15100
15050 B1=1
15100 IFZE=>9ANDZE<11THENEE=EE+1:IFEE=1THENPRINT"DU MUSST WAS ESSEN":ES=1:HH=1
15110 IFZE=>11ANDES=1THEN46000
15120 IFZE=>12THENEE=0
15200 IFZE=>15ANDZE<17THENTT=TT+1:IFTT=1THENPRINT"DU MUSST WAS TRINKEN":TR=1:HH=
1
15210 IFZE=>3ANDZE<5THENTT=TT+1:IFTT=1THENPRINT"DU MUSST WAS TRINKEN":TR=1:HH=1
15220 IFZE=>17ANDTR=1ORZE=>5ANDZE<12ANDTR=1THEN46100
15230 IFZE=>13ORZE=>6ANDZE<7THENTT=0
15240 IFZE=18THENGOSUB20000
15250 IFZE=20THENKS=0:KL=0:SE=0:SZ=0
15300 IFZE=>20ANDZE<22THENPP=PP+1:IFPP=1THENPRINT"DU MUSST SCHLAFEN":PE=1:HH=1
15310 IFZE=>22ANDPE=1THEN46200
15320 IFZE>22THENPP=0
15400 IFZE<>18THEN15420
15405 KK=KK+1:IFKK<>1THEN15420
15410 KE=1:SH=1
15412 X=INT(RND(1)*2)+1:IFX=1THENKS=1:GOTO15416
15414 KL=1
15416 X=INT(RND(1)*2)+1:IFX=1THENSE=1:GOTO15420
15418 SZ=1
15420 IFKE=2ANDSH=2THENLI=1
15430 IFZE=9THENKE=0:SH=0:LI=0:KK=0
15500 IFGE=1THENHP=HP+1:IFHP<>1THEN15600
15505 IFGE=0THEN15600
15510 X=INT(RND(1)*2)+1:IFX=1THENPF=1:GOTO15520
15515 PH=1
15520 X=INT(RND(1)*2)+1:IFX=1THENHA=1:GOTO15600
15530 HB=1
15600 IFDR=1ANDGG=1THEN46700
15700 IFVN$="WARTE"THENWA$=TI$:WAIT197,64:WAIT197,32:TI$=WA$:H=1
15950 IFHH=1THENFORQ=1TO2000:NEXTQ:HH=0
15990 IFSL=1THENNEXT:TI$="000000"
15999 Ki=0:JJ=0:II=0:RETURN
20000 LU=LU+1:IFLU>1THENRETURN
20010 IFZE=20THENLU=0
20020 PRINT"ES WIRD DUNKEL":HH=1:RETURN
20100 FORQ=180TO130STEP-1:POKE36878,15:POKE36874,Q:POKE36875,Q:POKE36876,Q:NEXTQ
20110 FORQ=255TO220STEP-1:POKE36878,15:POKE36874,Q:POKE36875,Q:POKE36876,Q:NEXTQ
20120 FORQ=0TO4:POKE36874+Q,0:NEXT:RETURN
20200 FORQ=1TO10:POKE36878,15:POKE36874,230:FORQQ=1TO30:NEXTQQ
20210 POKE36878,15:POKE36874,240:FORQQ=1TO30:NEXTQQ,Q
20220 FORQ=0TO4:POKE36874+Q,0:NEXT:RETURN
20300 FORQQ=1TO5
20310 FORQ=220TO240:POKE36878,15:POKE36874,Q:NEXTQ,QQ:POKE36874,0:RETURN
20400 FORQQ=250TO128STEP-5:POKE36878,15:POKE36874,QQ:FORJ=1TO30:NEXTJ
20410 POKE36874,0:FORJ=1TO30:NEXTJ,QQ:RETURN
21000 PRINTQ$"DA IST EIN HAMMER":H=1:RETURN
21010 PRINTQ$;"DA IST EINE LUKE":H=1:RETURN
21020 PRINTQ$;"DA IST EIN HEBEL":H=1:RETURN
21030 PRINTQ$"DA IST EIN SCHLUESSEL":H=1:RETURN
21040 PRINTQ$"DA IST EIN PFLOCK":H=1:RETURN
21100 GOSUB49000:VN$=""
21105 GETA$:IFA$=""THENGOSUB21200:GOTO21105
21110 PRINTQ$" "A$;:INPUT"[illegible]";VN$:VN$=A$+VN$:RETURN
21200 ZU=ZE+VAL(MID$(TI$,3,2))
21210 PRINT"[illegible]ES IST"ZU"[illegible]."RIGHT$(TI$,2)" UHR "Q$"[illegible]":RETURN
40000 REM
40010 PRINTQ$;"RICHTUNG NICHT "
40015 PRINT"MOEGLICH"
40030 RETURN
40100 PRINTQ$"EINGANGSTUER IST"
40110 PRINT"ZUGESCHLOSSEN"
40130 RETURN
40200 PRINTQ$"BEFEHL FALSCH":RETURN
40300 IFLI=1THENPRINT"[illegible]"
40310 IFLI=0THENPRINT"[illegible]"
40330 PRINT"[illegible] "
40340 PRINT"[illegible] "
40341 PRINT"[illegible] "
40342 PRINT"[illegible] "
40343 PRINT"[illegible]"
40390 RETURN
40400 REM
40410 PRINTQ$;"TUER IST ZUGESCHLOS-"
40420 PRINT"SEN"
40430 FORQ=1TO2000:NEXT:ZL=ZL+1:RETURN
40500 PRINTQ$;"WOMIT ?":RETURN
40600 PRINTQ$;"IST GEOEFFNET":RETURN
40700 PRINTQ$;"OK HAST GENOMMEN":H=1:RETURN
41000 PRINTQ$;"DA IST NICHTS VON":PRINT"INTERESSE":H=1:RETURN
41100 PRINT"[illegible]TUER IST ZUGESCHLOS-"
41110 PRINT"SEN"
41120 H=1:ZL=1:ONYYGOTO4900,6900,3900
41500 X=INT(RND(1)*3)+1
41520 AD=ZE:SS=1
41530 FORJK=1TOX:AD=AD+1:IFAD=>24THENAD=0
41540 NEXT
41580 PRINTQ$;"[illegible]SUCH DEN SCHLUESSEL"
41585 PRINT"IN"X"STUNDEN"
41590 RETURN
```

```
45000 PRINT"♥"
45001 GOSUB21200:VN$=""
45002 IFLI=1THENPRINT"▒▒"
45003 IFLI=0THENPRINT"▒■"
45005 FORI=1TO13
45010 PRINT"▒";Z$(ZI,I)
45020 NEXT:PRINT"                    "
45025 GOSUB14000::GOSUB12000:H=0:RETURN
46000 PRINT"♥▒▒▒DU HAST VERLOREN       ▒▒▒DU BIST VERHUNGERT":GOSUB20400:END
46100 PRINT"♥▒▒▒DU HAST VERLOREN       ▒▒▒DU BIST VERDURSTET":GOSUB20400:END
46200 PRINT"♥▒▒▒DU HAST VERLOREN       ▒▒▒DU BIST VOR MUEDIGKEIT"
46210 PRINT"▒TOD UMGEFALLEN":GOSUB20400:END
46300 PRINT"♥▒▒DU HAST VERLOREN"
46305 PRINT"▒▒DER GEHILFE DRACULAS  ▒HAT DICH GETOETET":GOSUB20400:END
46400 PRINT"♥▒▒▒DU HAST VERLOREN"
46405 PRINT"▒▒DRACULA HAT DICH       ▒GEBISSEN":GOSUB20400:END
46700 PRINT"♥▒▒▒DU HAST GEWONNEN       ▒DU HAST JANE BEFREIT":POKE36875,0:POKE368
76,0
46710 FORQ=1TO3:FORQQ=160TO250STEP20:POKE36878,15:POKE36874,QQ:FORJ=1TO80:NEXTJ,
QQ
46720 POKE36874,0:FORJ=1TO80:NEXTJ:FORQQ=240TO200STEP-20:POKE36874,QQ
46730 FORI=1TO80:POKE36879,I:NEXTI,QQ,Q:POKE36874,0:POKE36879,27
46790 END
47000 ZT$=TI$:TI$="000000":GOSUB49000:POKE36877,140:POKE36874,128
47005 PRINTQ$;"DU MUSST NOCH"TI%"▒ "
47010 IFTI=>7200THENTI$=ZT$:ZE=ZE+2:ZN=ZN+2:POKE36877,0:POKE36874,0:RETURN
47012 POKE36878,0:IFPEEK(162)<64THENPOKE36878,PEEK(162)/4
47020 TI%=120-TI/60:PRINTQ$;"▒▒MINUTEN SCHLAFEN":GOTO47005
49000 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"
49010 FORI=1TO7
49020 PRINT"                    "
49030 NEXT:RETURN
50000 Z$(1,1)="   @ABC!!!!!!:▔DEFG  "
50005 Z$(1,2)="ABC     !      ┌    DE"
50010 Z$(1,3)="        !      ┌      "
50015 Z$(1,4)="        !  ♥!┐ ┌      "
50020 Z$(1,5)="   HI!  !  !┌ ┌ ┌JK   "
50025 Z$(1,6)="    R!  !  !┌ ┌ ┌  L  "
50030 Z$(1,7)="    Q!  !  !┌ ┌ ┌  ┌  "
50035 Z$(1,8)="   ON!  ───●─● ┌  ┌  "
50040 Z$(1,9)="   MN▒        ■ ■ ┌  "
50045 Z$(1,10)="    !▔          ■ ■┌ "
50050 Z$(1,11)="     ▒           ■ ┬ "
50055 Z$(1,12)="    ▔             ■  "
50060 Z$(1,13)="  ▒                ■ "
50100 Z$(2,1)="   @ABC♥!! !┌!:▔DEFG  "
50105 Z$(2,2)="ABC▔▔▔▔▔!  !┌  ┌   DE"
50110 Z$(2,3)="▔▔▔▔▔▔▔▔!  !┌  ┌      "
50115 Z$(2,4)="▔▔▔▔▔▔▔▔!  !┌  ┌ ┌JK  "
50120 Z$(2,5)="▔▔▔▔▔▔▔▔! ▔──┤ ┌ ┌  L "
50125 Z$(2,6)="▔▔▔▔▔▔▔▔! !♥┬ ┌ ┌  ┌  "
50130 Z$(2,7)="▔▔▔▔▔▔▔▔! !!┌┌ ┌ ┌  ┌  "
50135 Z$(2,8)="▔▔▔▔▔▔▔!!!!!!▔■■ ■ ┌  "
50140 Z$(2,9)="▔▔▔▔▔▔▔  ──────┌┌ ■ ■┌ "
50145 Z$(2,10)="▔▔▔▔▔▔ ▒       !┬  ■ ┬ "
50150 Z$(2,11)="▔▔▔▔▔ ▒        ! !  ■  "
50155 Z$(2,12)="▔▔▔ ┌┐!!!!!!┬┬┐     ■  "
50160 Z$(2,13)="▔▔  ┌    ─    ┌      ■ "
50200 Z$(6,1)="            ▒■        "
50205 Z$(6,2)="           ▒   ■      "
50210 Z$(6,3)="          ▒     ■     "
50215 Z$(6,4)="         ▔──────┤     "
50220 Z$(6,5)="        ▒/       └■   "
50225 Z$(6,6)="      ▒     ♥┐  ♠■    "
50230 Z$(6,7)="     ▒      ─●  +  ■  "
50235 Z$(6,8)="    ▒           ◥  ■  "
50237 Z$(6,9)="   ▒  O    ──        ■ "
50240 Z$(6,10)="  ▒  ♥┐  ┌──!      ·  ■"
50245 Z$(6,11)="  !  !┌  ┌──├──┤  ───┌"
50250 Z$(6,12)="  !  ♥┌  O──! ♥┐  ■─▒┌"
50255 Z$(6,13)="  !  !┌    !! O─●  ▔─┤┌"
50300 Z$(4,1)="    @ABC♥!!!!▔DEFG   "
50305 Z$(4,2)="@ABC  ▪ ┘ !    ┌▪ ┘ DEF"
50310 Z$(4,3)=" ┘   ▪┘  !     ┌▪┘  ▪  "
50315 Z$(4,4)="     ▪ ┘┘▪!    ┌  ┘     "
50320 Z$(4,5)=" ┘   ┘  ▪ !    ┌▪  ┘   "
50325 Z$(4,6)="      ▪ ┘▪! ── ┌┘   ▪  "
50330 Z$(4,7)="   ┘  ▪ ┘!┌─!┌ ┘┘  ▪   "
50335 Z$(4,8)=" ▪   ┘ ┘▪!┌┬S┌ ┘  ▪┘ ┘ "
50340 Z$(4,9)=" ┘    ▪   _!┌┬┬●  ┘  ▪ ▪"
50345 Z$(4,10)="  ┘     ▒ !┌▒   ■▪    ┘"
50350 Z$(4,11)=" ┘   ▪▪ ▒  !!     ■ ┘ ▪ "
50355 Z$(4,12)=" ┘   ▪ ▒           ■▪┘ ▪"
50360 Z$(4,13)="▪ ┘ ▒               ■┘▪ "
50400 Z$(5,1)="   @ABC♥!!!!!!▔DEFG  "
50405 Z$(5,2)="ABC      :      ┌    DE"
50410 Z$(5,3)="      U  !   ▒▒▒┌      "
50415 Z$(5,4)="     HI┌  !♥┐ ▒▒▒┌      "
50420 Z$(5,5)="  HV   ┌ !┌    ┌       "
50425 Z$(5,6)="  !┌   ┌ !!┌   ─●──    "
50430 Z$(5,7)="  !┌   ▒ !!┌    ! ┌S   "
50435 Z$(5,8)="  !┌ ▒ ▒!!!!!!!! !  ┌┬  "
50440 Z$(5,9)="  !┌▒ ▒       ┌▒!!!JK  "
50445 Z$(5,10)="  !┌ ▒       /X/X ■    JK"
50450 Z$(5,11)="  !┌▒         /X/X ■    "
50455 Z$(5,12)="  !▒           /X/X ■───"
50460 Z$(5,13)="  ▔             /X/X  ! "
50500 Z$(3,1)="           ──          "
50505 Z$(3,2)="          ▒ /■         "
50510 Z$(3,3)="         ♥     ┐       "
50515 Z$(3,4)="        ▒       ■      "
50520 Z$(3,5)="       ♥         ┐     "
50525 Z$(3,6)="      ▒           ■    "
50530 Z$(3,7)="     ♥   /─────────┐   "
50535 Z$(3,8)="    ▒    ──────────  ■  "
50540 Z$(3,9)="   ♥     ─────────X   ┐ "
50545 Z$(3,10)="  ▒  ♥┐              ♥┐ ■"
50550 Z$(3,11)="  !  !┌     ────     !┌ ┌"
50555 Z$(3,12)="  !  !!  ▒┘┘┘┘■     !┌ ┌"
50560 Z$(3,13)=" /!  !┌  ♥▪!!!▪!    !┌ ┌"
50600 Z$(7,1)="          ──           "
50605 Z$(7,2)="   ─♠!─▔─┘ / !─▔─!♠─   "
50610 Z$(7,3)="─┘!───────────────────┘"
50615 Z$(7,4)="───────────────────────"
50620 Z$(7,5)="───────♠!─▔─!♠──────── "
50625 Z$(7,6)="──────┌!         !──────"
50630 Z$(7,7)="─────/             \────"
50635 Z$(7,8)="─────\             \────"
50640 Z$(7,9)="─────┴             /────"
50645 Z$(7,10)="─────/              └────"
50650 Z$(7,11)="─────\ ──     ──  \────"
50655 Z$(7,12)="─────\┐──┤  ▔──┤/──────"
50660 Z$(7,13)="─────  ■──▒X■──▒┌──────"
50700 Q$="▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"
60000 RETURN

READY.
```

Starship VC-20




128 129 130 131 192 193 194 195
132 133 134 135 196 197 198 199
136 137 138 139 200 201 202 203
140 141 142 143 204 205 206 207
144 145 146 147 208 209 210 211
148 149 150 151 212 213 214 215
152 153 154 155 216 217 218 219
156 157 158 159 220 221 222 223
160 161 162 163 224 225 226 227
164 165 166 167 228 229 230 231
168 169 170 171 232 233 234 235
172 173 174 175 236 237 238 239
176 177 178 179 240 241 242 243
180 181 182 183 244 245 246 247
184 185 186 187 248 249 250 251
188 189 190 191 252 253 254 255

VC-261/HC

- der Zeichengenerator

Zweiter Teil

Viele Telefonanrufe erreichten die Redaktion mit Fragen zum Zeichengenerator. Deshalb haben wir die VC-20-Columne kurzfristig umgestaltet. Die Invader-Invasion findet darum erst in einem Monat statt.

Eigentlich klar, daß viele Neulinge, die erst seit einigen Monaten selbst programmieren, nicht mit den Erklärungen zum Zeichengenerator auskamen. Man muß nämlich über einiges zusätzliches Wissen und über eins oder meherere zusätzliche Handbücher verfügen, um sich die auftretenden Ungereimtheiten erklären zu können.

POKE44,28:POKE7168,0:NEW

Bevor der Zeichengenerator eingetastet werden kann, muß der Basic-Start auf eine höhere Stelle im RAM verschoben werden. Denn sonst würde der neue Zeichensatz das soeben eingetastete Programm gleich wieder zerstören. Das geschieht mittels der Befehle: POKE 44,28 : POKE 7168,0 : NEW
Das NEW ist nicht notwendig, wenn man den Zeichengenerator von Diskette oder von Kassette lädt; nur vor dem ersten Eintasten muß es durchgeführt werden.

POKE43,227:POKE44,19:POKE5090,0

Wie arbeitet man nun mit dem selbst definierten Zeichensatz.
Setzen wir einmal voraus, daß soeben ein Zeichensatz neu definiert wurde und dieser wurde zur Sicherheit auf einer Kassette abgelegt.
Um das neue Programm einzutasten, löschen wir den Zeichengenerator mit NEW. Das neue Programm beginnen wir mit der Zeile:

5 POKE 36869,205
Hierdurch wird das System auf den neuen Zeichensatz eingestellt.
Nach Eingabe des neuen Programms wird selbiges auf Kassette (oder Disk) geSAVEd. Auf dem Datenträger befindet sich nun eine Version ohne die neuen Zeichen!
Dann wird mittels:

POKE 43,227: POKE 44,19

die Vorbereitung getroffen, das Programm noch einmal zu retten und zwar diesmal mit dem neuen Zeichensatz. Auf unserer Kassette befindet sich nun eine Programmversion ohne und eine mit neuem Zeichensatz. Zwei Versionen sind besser, weil man dann die Zeichen mit dem Zeichengenerator noch verbessern und später mit dem Programm wieder verbinden kann.
Nach dem SAVE einfach RUN eingeben und das kurze Programm vor dem Zeichensatz stellt den Basic-Start-Zeiger wieder an die Stelle, an der das neue Programm anfängt.
Noch einige Erläuterungen zu den Kommandos:
<=> Dupliziere Zeichen
Mittels einer Zahl zwischen 0 und 255, gefolgt von Druck auf die Taste RETURN wird ja das aktuelle zu bearbeitende Zeichen bestimmt. Will man sich die Mühe sparen jedes Zeichen neu zu definieren (äußert unsinnig, wenn man z.B. ein A und ein Ä benötigt), so gibt man die Zahl für das Zeichen das zu erstellen ist, ein (in unserem Beispiel also die Zahl des Zeichens, an der das Ä stehen soll). Danach drückt man die Taste = und es erscheint das Wort WIRD. Nun wird wieder eine Zahl eingegeben und zwar die Zahl des Zeichens, dem das neue Zeichen ähneln soll. (in unserem Beispiel also eine 1 gefolgt von RETURN für das große A).
Das Zeichen kann nun abgeändert werden.

Homecomputer's Service-Telefon

Wir wissen, wie ärgerlich es ist, wenn Programme, die man aus Zeitschriften in mühsamer Kleinarbeit abtippt, nicht funktionieren.
Leider ist man gegen Bugs nie gefeit. Es kann dermaßen viel passieren, bis das fertige Programm zu Ihnen, dem Leser kommt. Beim Layout können Zeilen verloren gehen; oder wenn gerade für das eine oder andere Gerät kein Drukker im Hause ist, das Programm aber veröffentlicht werden soll, man es also in ein anderes Gerät eintastet und über dessen Drucker ausgibt, können sich leicht Tippfehler ein schleichen, die auch bei wiederholten Korrekturlesen nicht entdeckt werden. Und, und, und ...
Natürlich geben wir uns alle erdenkliche Mühe, Fehler auszuschließen, aber wie schon gesagt, es kann halt einiges passieren.
Deshalb werden wir als zusätzlichen Leserservice für alle Fälle in den nächsten Tagen einen Telefon-Service für diese Fälle einrichten. Im nächsten Heft geben wir Ihnen die Telefonnummer bekannt, unter der Sie eventuelle Programmfehler erfahren können.
Außerdem werden in jedem Falle im folgenden Heft nötige Programmkorrekturen abgedruckt.

Assembler für VC-20 & VC-64

Teil zwei unserer Programm-Serie zum Programmieren von Unterprogrammen in Maschinensprache ist der Assembler.

Nach dem Disassembler, den wir in Heft 3/83 veröffentlichten folgt nun der Assembler, den im nächsten Heft der Editor folgen wird.
In den Zeilen 20000-20030 befindet sich eine Routine, mit der getestet werden kann, ob alles richtig eingetastet wurde.
Dieses Testprogramm wird mit RUN 20000 gestartet. Zuvor wird jedoch eine leere Kassette in den Recorder gelegt. Man hat jetzt die Möglichkeit, Programmzeilen eines Assembler-Programms mit folgender Syntax einzugeben:

"LABEL MNEMONIC OPERAND

Zum Beispiel:

```
"VIDMAT EQU 4096
" ORG $6000
"START LDA #0
"NXTZCHN LDX #0
"NXTPOS STA VIDMAT,X
" INX
" BNE NXTPOS
" CLC
" ADC #1
" BNE NXTZCHN
" RTS
END
```

Danach wird das Programm mit RUN gestartet und das Testprogramm assembliert. Es handelt sich um einen Zweipass-Assembler, also muß die Kassette nach dem ersten Durchlauf nochmals zurückgespult werden um in zweiten Durchlauf das Maschinenprogramm in seiner entgültigen Form in die Speicherzellen des VolksComputers zu POKEn.

Das Assembler-Listing ist ein extrem simples Programm, das nichts weiter tut, als die ersten 256 Zeichen der Video-Matrix mit einem Durchlauf des Commodore-Zeichensatzes zu versorgen. Sofern sich an dieser Stelle in der Farbmatrix keine anders lautende Information befindet als die Hintergrundfarbe, geschieht allerdings nichts sichtbares. Es ist daher gut, wenn der obere Bildschirmbereich vorher mit irgendwelchen Zeichen vollgeschrieben wird, bevor man das
SYS 24576 aufruft, das das Maschinenprogramm startet.

Ein Komfortabler Editor folgt, wie bereits angesprochen im nächsten Heft, wo auch die Verknüpfung der einzelnen Programmteile erklärt wird.

```
100 READO$:READA$:O$=O$+A$:READA$:O$=O$+A$
110 FORI=0TO7:READAI$(I):NEXT
120 DIMO%(58):FORI=0TO55:READO%(I):NEXT
130 FORI=0TO7:READS%(I):NEXT
200 GOSUB50000
500 OPEN1,1:Z=0
510 Z=Z+1:INPUT#1,L$:FORI=1TO9:IFMID$(L$,I,1)<>" "THENNEXT
520 IFI-1THENA$(LC)=LEFT$(L$,I-1):A(LC)=PC:LC=LC+1
525 PRINTPC,L$
530 GOSUB4000
540 PC=PC+L
900 IF(64ANDST)=0THEN510
910 CLOSE1
1000 GOSUB50000
1010 OPEN1,1:F=-1:Z=0
1020 Z=Z+1:INPUT#1,L$:FORI=1TO9:IFMID$(L$,I,1)<>" "THENNEXT
1030 GOSUB4000:IFOI>55THEN1150
1040 O=O%(OI):IFOI>21THEN1070
1050 IFAI=1ANDOAND1ORL=3THENO=O+8
1060 O=O+S%(AI)
1070 GOSUB5000:IFOI>46ANDOI<55THENS=S-PC-2:S=S+(256ANDS<0)
1080 S%=S/256:S=S-256*S%:PRINTPC,O,S,S%
1090 POKEPC,O:POKEPC+1,S:IFL=3THENPOKEPC+2,S%
1100 PC=PC+L:IF(64ANDST)=0THEN1020
1110 CLOSE1
1120 END
1150 IFOI-57THEN1100
1160 K=0:FORP=PCTOP+L-1:A$="":FORK=K+1TOLEN(L$):J$=MID$(L$,K,1)
1165 IFJ$<>","THENA$=A$+J$:NEXT
1170 GOSUB5000:POKEP,S:NEXTP:GOTO1100
```

Hallo Computer Club bitte melden

Eine große Anzahl von Computerclubs gibt es in Deutschland, Österreich und der Schweiz, den Ländern, in denen Homecomputer verbreitet wird, bereits. Aber wo sich gerade neue Clubs gründen, welche Ziele die einzelnen Clubs verfolgen, um welche Computertypen es sich handelt, die in diesen Clubs genutzt werden, das können wir nur duch Zuschriften von Clubmitgliedern dieser Clubs erfahren.

Nach unserem ersten Artikel über eine Regionalgruppe der Apple User Group erreichten uns jede Menge Telefonanrufe und Zuschriften mit Anfragen nach Computerclubs in der Nähe dieser Interessenten.

Wir können natürlich nicht jedesmal eine vollständige Liste abdrucken, in der säntliche Clubs verzeichnet sind; der Platz den wir für die Computer Club Rubrik zur Verfügung haben ist beschränkt; z. Zt. auf 2 - 3, in einigen Monaten vielleicht auf 4 - 5 Seiten. Doch möchten wir schon den einzelnen Clubs Gelegenheit geben, ihre Adressen und einige Informationen zu verbreiten. Auf jeden Fall soll der Platz sinnvoll genutzt werden, mit Berichten über Aktivitäten der Clubs, über Neugründungen und soweit es uns möglich ist, Aktivitäten zu unterstützen, damit sie mehr Resonanz erzielen.

Computer Club Freunde, dies ist Ihre Rubrik. Schreiben Sie uns, was in Ihrem Club passiert, was geplant ist und wie Mitglieder und Außenstehende auf Aktionen und Ideen reagieren.

```
4000 A$=MID$(L$,I+1,3):L=0
4010 FORJ=1TO175STEP3:IFA$<>MID$(O$,J,3)THENNEXT:PRINT"OPCODE ERROR IN"Z
4020 OI=(J-1)/3:L$=MID$(L$,I+5,30)
4030 IFOI=56ANDNOTFTHENA$=L$:GOSUB5000:A(LC-1)=S
4032 IFOI=57THENL=1:FORJ=1TOLEN(L$):L=L-(MID$(L$,J,1)=","):NEXT
4035 IFOI=58THENA$=L$:GOSUB5000:PC=S
4040 IFOI>55THENRETURN
4050 IFLEN(L$)=0ORL$="A"THENB$="A":GOTO4140
4060 A$="":B$="":FORI=1TOLEN(L$):J$=MID$(L$,I,1)
4070 IFJ$="#"ORJ$="("THENB$=B$+J$:GOTO4130
4090 IFJ$=")"THEN4130
4100 IFJ$=","THENI=I+1:B$=B$+MID$(L$,I,1):GOTO4130
4120 A$=A$+J$
4130 NEXT:L=-(OI>21)-(OI>46)-(OI>54):IFLTHENRETURN
4140 FORAI=0TO7:IFB$<>AI$(AI)THENNEXT:PRINT"OPER ERROR IN"Z
4150 L=1-(AI>0)-(AI>3):IF(OI=12OROI=15)ANDAI=4THENAI=7
4155 IFAI<6THENRETURN
4160 GOSUB5000:IFE=1OROI=7THENL=3:RETURN
4170 L=2-(S>255):RETURN
5000 E=0:S=VAL(A$):IFSTHENRETURN
5010 IFASC(A$)<>36THEN5030
5020 FORJ=2TOLEN(A$):J$=MID$(A$,J,1):S=16*S+VAL(J$)+(ASC(J$)-55ANDJ$>"@"):NEXT:R
ETURN
5030 IFASC(A$)=37THENFORJ=2TOLEN(A$):S=S+S+VAL(MID$(A$,J,1)):NEXT:RETURN
5040 FORJ=0TOLC-1:IFA$=A$(J)THENS=A(J):RETURN
5050 NEXT:E=1:RETURN
10000 DATAORAASLBITANDROLEORLSRJMPADCRORSTYSTASTXLDYLDALDXCPYCMPDECCPXSBCINC
10010 DATABRKPHPCLCPLPSECRTIPHACLIRTSPLASEIDEYTXATYATXSTAYTAXCLVTSXINYDEXCLDINXN
OP
10020 DATASEDBPLBMIBVCBVSBCCBCSBNEBEQJSREQUDFBORG
10030 DATAA,#,(X,(Y,Y,(,"",X
10040 DATA5,6,36,37,38,69,70,68,101,102,132,133,134,164,165,166,196,197,198,228,
229
10050 DATA230,0,8,24,40,56,64,72,88,96,104,120,136,138,152,154,168,170,184,186,2
00
10060 DATA202,216,232,234,248,16,48,80,112,144,170,208,240,32
10070 DATA4,-4,-4,12,12,32,0,16
20000 OPEN1,1,1
20010 INPUTA$
20020 IFA$><"END"THENPRINT#1,CHR$(34);A$:GOTO20010
20030 CLOSE1:END
30000 OPEN1:FORI=1TO10:INPUT#1,A$:PRINTA$:NEXT:CLOSE1
40000 INPUTA$:A$="$"+A$:GOSUB5000:PRINTS:GOTO40000
50000 PRINT"█CASSETTE ZURUECKSPULEN█DANN █ P █ DRUECKEN"
50005 GETWA$:IFWA$><"P"THEN50005
50010 PRINT:RETURN
```

Cornelius Geppert (Ge)

1.Computertag in Wattenscheid

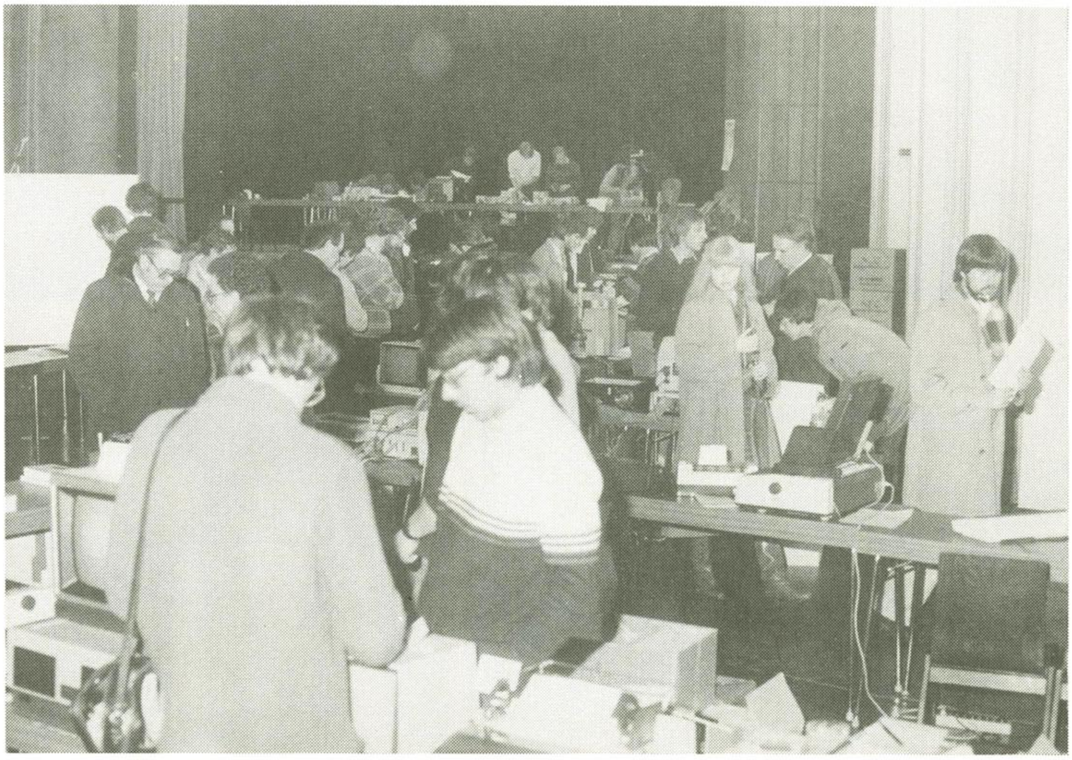


Wie erfolgreich die Aktivitäten einiger Computerclubs sind, lesen Sie in unserem Bericht über die Veranstaltung eines Computertages mit Beteiligung namhafter Firmen.

Was sich am 27.2.83, einem Sonntag vormittags vor der Stadthalle in Wattenscheid ereignete, haben selbst die optimistischen Verstalter nicht erwartet. Hunderte von Computerfans, aus weiten Bereichen des Bundesgebiets angereist, standen Schlange, um Eintrittskarten für den ersten Wattenscheider Computertag zu erhalten.

Unter der Schirmherrschaft der Apple User Group hatten Mitglieder des Vereins eine schier unmöglich scheinende Leistung vollbracht. Es war nicht nur gelungen, bekannte Firmen wie Basis, IBS, Bartscher Elektronik, Hoco und Unitronic dazu zu bewegen, ihre Produkte vorzustellen, sondern dieses auch noch überregional bekannt zu machen.

Wie uns einer der Veranstalter, Herr Giese, im Gespräch mitteilte, hätten die ersten Wochen der Planung keineswegs erkennen lassen, daß sich eine solch aktraktive Schau aus der Idee entwickeln würde, weil zunächst kein Interesse seitens der Industrie zu verzeichnen war. Erst nach unzähligen Telefonaten und nach enormen persönlichen Engagement war es dann doch gelungen, professionelle Aussteller zur Teilnahme zu bewegen, ohne die kaum ein solches Interesse der Besucher hätte geweckt werden können.



Obwohl natürlich die einzelnen Aktionsgruppen des Vereins ihre Arbeiten vorstellten - immer wieder interessant GALA, die Sprache zur Spielentwicklung - hatten schließlich die Profianbieter wie erwartet den größten Andrang zu verzeichnen.

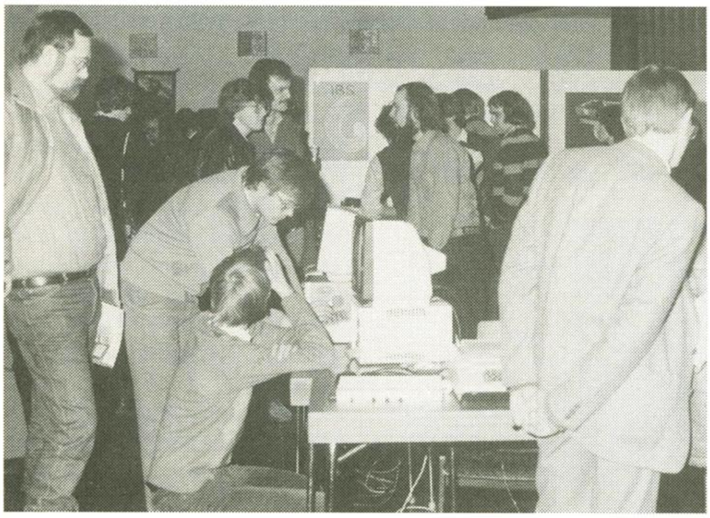
Ständig umlagert waren der Unitronicstand, wo der Besucher Gelegenheit hatte, den Multitech Microprofessor II näher kennenzulernen, der Stand von Basis, wo Wissenwertes zum 108 vermittelt wurde, und auch die Apple-



Interface-Entwickler IBS und Bartscher Elektonik fanden reges Interesse an ihren Karten vor.
Wer zum Schauen gekommen war, kam in jeden Falle auf seine Kosten und auch diejenigen, die mal soeben ihren Apple um eine Speichererweiterung, eine Z80-Karte oder die 80-Zeichenkarte erweitern wollten, nutzten die Gelegenheit des Tages, da es an mehreren Ständen Computertag-Rabatte gab.
Am Abend waren jedenfalls alle zufrieden über diesen gelungenen Tag; Veranstalter, Aussteller und Besucher. Und auch wir, denn für uns war es die erste Gelegenheit (einen Tag vor dem Erstausgabetag!), im persönlichen Eindruck zu testen, wie unser Magazin beim Leser von Computerzeitschriften aufgenommen würde.

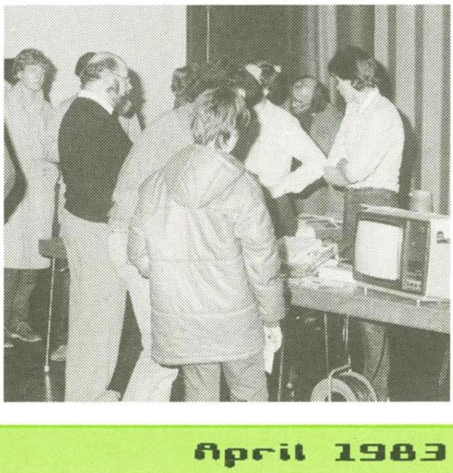
Pictures of an Exhibition

(Bilder einer Ausstellung)









Biete an Software

Apple II: Software auf Cassette! Spallek Eiswaldtstr. 21d 1 Berlin 46

VC-20 Programme ab 1 DM!!! Auch Tausch! Info: 80 Pf bei W. Kratz, Benediktinerweg 23, 5100 Aachen 1

ZX-SPECTRUM ZX81 ZX-SPECTRUM
● Superspiele u. vieles andere ●
● Info 80 Pf Dipl.-Ing. G. Verse ●
● 465 Gelsenk., Grüner Weg 45 ●

******* SUPER-VEW ******* Einmaliges EXBASIC-LEVEL-II-Prg. zur Auswertung sämtlicher VEW-Systeme (6/49 & 7/38). Info von Th. Preymesser, Hesselbergring 19, 8500 Nürnberg 60.

Die neuesten Programme aus England für ZX81, Acorn Atom, VC20, Spectrum und BBC. Schach, FORTH, Spiele u.a. Katalog gegen 1,80 DM in Briefm. T. Wagner, Softwarevers. Postfach 112243 G, 8900 Augsburg

Programme, Module+Zubehör (Drucker Diskettenlaufw.) für TI-99/4A, VC20, ZX81, MZ80A zu Sonderpreisen Info gg. DM 1: Boxner, Mittenfeld-Str. 50/2, 732 Göppingen 8

Biete an Hardware

VC-20, + Graphic 1211A + Literatur + 2 Joysticks (Atari) + Programme Cassetten zu verk. 040-6702929

VC-20, 8K, Datasette, Joysticks + Soft (Boss-Schach, Flugsimulator, Snackman, Asteroids, Biorh., 44-Z-Prog. + V.A.) + Bücher, auf alles 3 Monate Garantie; NP: über 1200,–, VB: 850. Tel. (0241) 26788 ab 18.00

VC-20: VC+DATASETTE+JOYSTICK + LIT + Software im Wert v. 1800 DM!!! VB 900 DM!!! Elfering 0251/33234 incl. 8K-Rammodule u! Modulbox!!

* MZ80K * 64KB Preis VHS: 07681/3835

Suche Software

Suche Software für VC20 Sharp 12 PC-1211/PC-1212/PC1251 Jörg Grenz Starnberger Str. 43 23 Kiel

Eilt! Wer hat Programm zur Auswertung einer Rally-Orientierungsfahrt für VC64? Walter Benndorf Rischbachstr. 1 6670 St. Ingbert

SUCHE VC64 PROGRAMME LISTE bitte an Peter Israel 1 Berlin 47, Goldammerstr. 22 030/6621205 auch Tausch, über 1000 Programme 80/3032

Verschiedenes

Verkaufe gebr. Spielcassette für ATARI VCS Systen H. Stiller, 6458 Rodenbach, Kastanienstr. 15

Nutzen Sie den Kleinanzeigenteil.

Homecomputer ist die Zeitschrift für die Anwendung der kleinen Microcomputer. Im Heim, in der Schule, im Club. Zum Spielen, Lernen und Verwalten.

Auftrag für Gelegenheitsanzeigen in Homecomputer

An Homecomputer
Gartenstraße 47
3440 Eschwege

Name und Adresse ______________________

______________________ Abo.-Nr. ______

Unterschrift Ich zahle sofort nach Rechnungserhalt. Datum

Unter der Rubrik „Kleinanzeigen“ veröffentlichen wir Gelegenheitsanzeigen für Verkaufsangebote, Kauf- und Tauschgesuche, Kontaktaufnahme bzw. Erfahrungsaustausch usw.

Preise für „Kleinanzeigen“: Private Gelegenheitsanzeige je Druckzeile 5,– DM inkl. MwSt. Chiffregebühr je Anzeige 10,– DM.

Gewerbliche Gelegenheitsanzeige je Druckzeile 7,– DM inkl. MwSt. (dürfen nicht unter Chiffre erscheinen).

Ich wünsche folgenden Text zu veröffentlichen:

Zutreffendes ankreuzen

suche ☐ Software — suche ☐ Hardware — ☐ Tausch — ☐ Versch.
biete an ☐ Software — biete an ☐ Hardware — ☐ Kontakte — ☐ Chiffre

250 Computer zu gewinnen





Eines dieser Geräte kann schon bald in Ihrem Hause stehen. Nur ein bischen Werbung für Homecomputer im Kollegen- und Freundeskreis und Sie können Ihren Traum-Computer gewinnen!

Daß Homecomputer anders ist als andere Computermagazine haben Sie schon bemerkt.

Allein in diesem Heft sind 16 Programme für die populärsten Microcomputer abgedruckt.

Darum haben wir uns auch etwas besonderes einfallen lassen, als es darum ging, eine Abonnenten-Werbeaktion zu starten.
Je mehr Leser Homecomputer regelmäßig beziehen, um so besser können wir das Magazin für Sie gestalten.
Denn Abonnenten bedeuten in unserem jetzigen Stadium mehr Mitarbeiter. Und mehr Mitarbeiter, das bedeutet, daß jeder mehr Sorgfalt in seine Programmierarbeit legen kann. Die Qualität steigt und das kommt Ihnen, dem Leser zugute.

Jeder, der einen Abonnenten für Homecomputer wirbt, erhält als kleines Dankeschön ein Taschenbuch zum Thema Microcomputer.

Außerdem verteilen wir 250 Microcomputer nach Wahl an diejenigen Leser, die Homecomputer die meisten Leser zuführen.

Die Gewinnverteilung erfolgt nach einem, für alle Leser interessanten Schlüssel:
Jedesmal, wenn bei uns 1000 neue Abonnenten registriert wurden, dann erhalten diejenigen 10 Leser, die zu diesem Zeitpunkt die meisten Abonnenten geworben haben - also in der Wettbewerbsliste vorne liegen, einen Microcomputer Ihrer Wahl; und zwar der Erste dieser Wettbewerbsliste einen Microcomputer im Werte von 2500 DM und die 9 Folgenden jeder einen im Werte von 1000,-DM.
Die 10 Gewinner werden aus der Wettbewerbsliste gestrichen (sie können allerdings wieder neu beginnen und nochmal gewinnen), und die ihnen Folgenden rücken nach.
Sind wieder 1000 neue Abonnenten registriert, dann geht es wie oben beschrieben weiter.
Teilnahmeberechtigt ist jeder, der Homecomputer regelmäßig bezieht.
Der Rechtsweg ist ausgeschlossen.
Die Gewinner werden sofort benachrichtigt und in der jeweils folgenden Ausgabe von Homecomputer bekanntgegeben.

End of File 4/83

Im nächsten Heft:

Super-Mondlandeprogramm für VC-64 und VC-20

Die bereits angekündigte Sprite-Grafik für Apple II

Action-Spiele und Utilities

5 TI-99/4A-Programme (Spiele, Heim, Utilities)

Wir blicken in die Tiefen des VC-64.
Tips und Tricks, um mehr aus dem VC-64 herauszuholen, als das Handbuch erahnen läßt.

Grandmaster - ein Schachprogramm für die Volkscomputer im Blickpunkt.

Wieder über 15 Programme für die populärsten Micros.

Programm-Fehler in 3/83

TI-99/4A:

In Zeile 500 muß stehen:
500 BY=FY
sonst läuft das Programm nicht.
In Zeile 870 gehört noch ein Blank.
Dann ist der Bildschirmausdruck schöner anzusehen.
870 DRUCK$="HIGHSCORE = "&ST R$(HS)

ZX-81:

In der "Zeile" 16586 muß es heißen:
16586 29 21 62 33 205 29 21 239

Verbesserungen:

In Apple-Programm EX-NK:
Nach 5 1000G einfügen:
6 FD12:7F EA EA
7 FE81:7F
8 F278:7F
Die ehemaligen Zeile 6-18 verschieben sich (9-21)

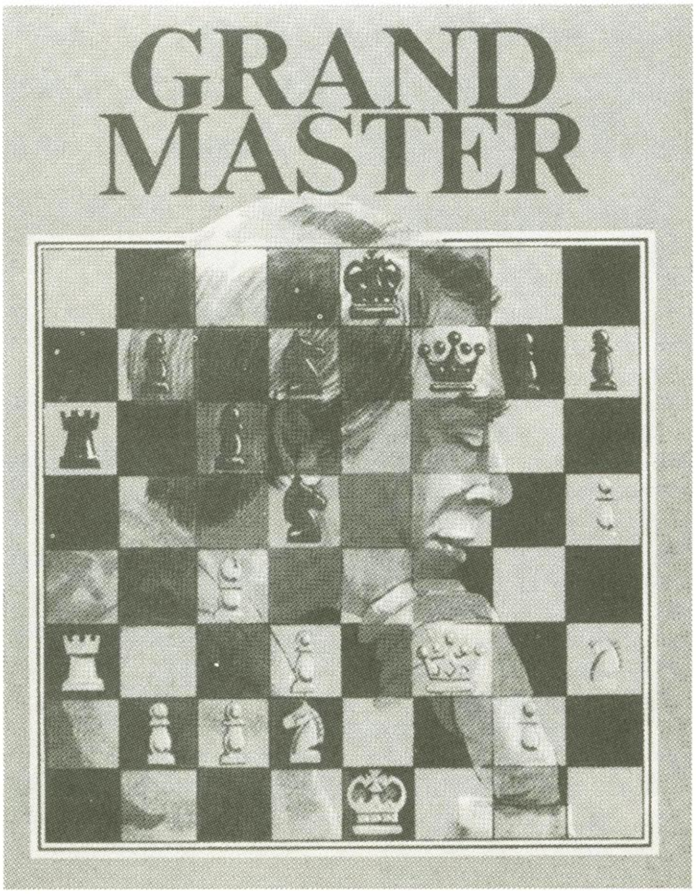
In VC-64-Programm Labyrinth:
Zeile 950: Der Wert für CD sollte besser =4 sein.

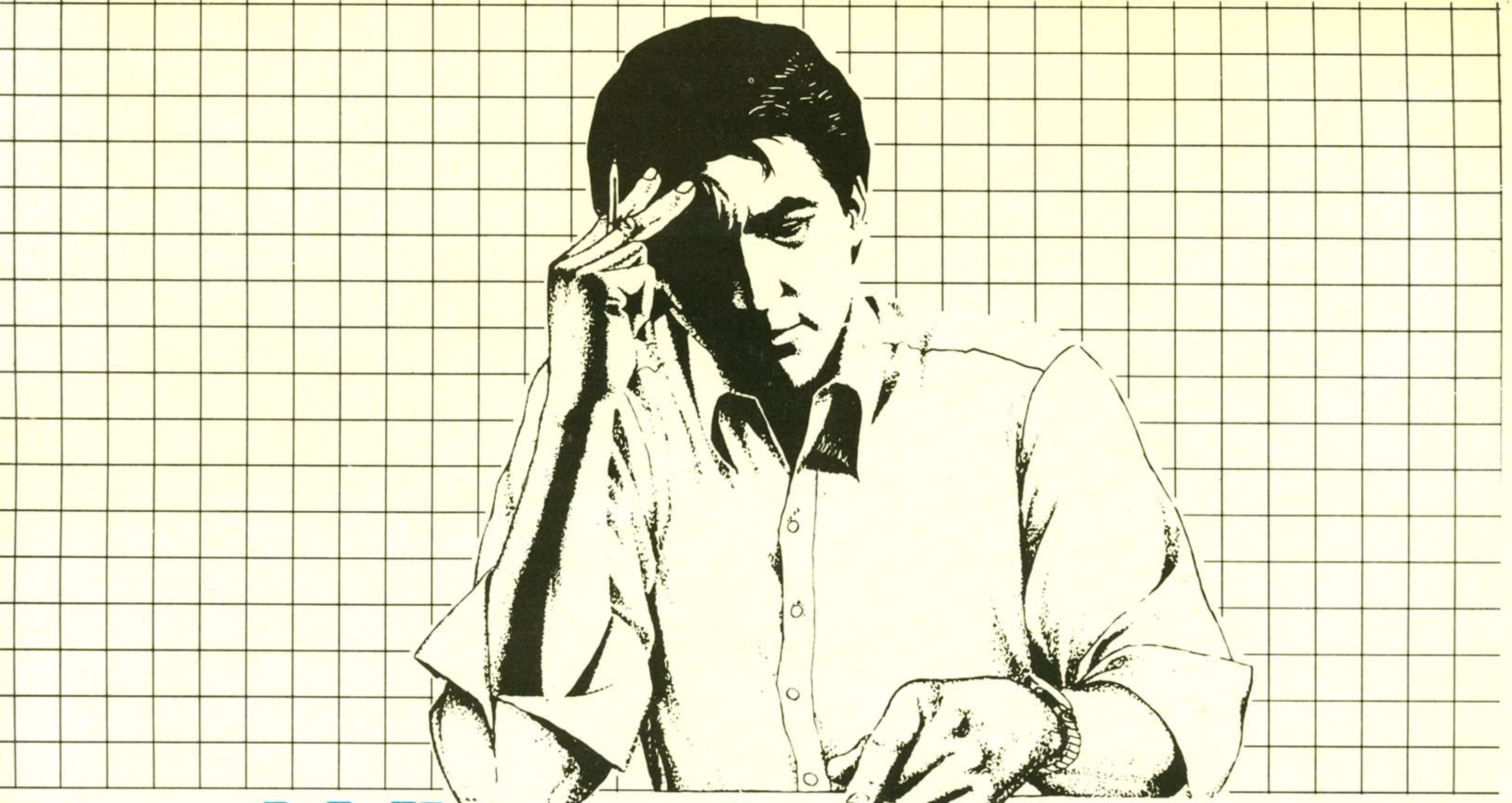
Leser-Software-Service

Nicht jeder Leser hat die Zeit und die Geduld, alle Programme, die er gerne hätte, in sein Gerät einzutasten.
Wir haben uns deshalb überlegt, daß es das einfachste ist, wenn wir die Programme, die wir ohnehin auf Diskette oder Kassette vorliegen haben, diesen Leserkreis zum Selbstkostenpreis (Datenträger, Kopierzeitaufwand, Porto, Verpackung und Mehrwertsteuer) anbieten.
Gegen Einsendung eines Schecks oder Vorauszahlung auf unser Konto bei der
Kreissparkasse Eschwege, Bankleitzahl 522 500 30
Nummer 45 22 934 senden wir Ihnen die gewünschten Programme schnellstmöglich zu.
Es genügt jeweils die Angabe eines Stichwortes, z.B.: VC-20-Disk Heft 4 oder Sinclair-Kassette Heft 4.

K = Kassette
D = Diskette

VC-64	K 8,-DM
...Assembler	
...Kalaha	
...Maze-Challenger	
CBM (Pet 3000)	K 8,-DM
Maze-Challenger	
Kalaha	
ZX-81	K 8,-DM
...Brüche pauken	
...Life	
...Survive	
...Labyrinth	
Apple II	D 15,-DM
...Suchrätsel	
...Disk-Schutz	
...Oktopus	
...Labyrinth	
VC-20	K 8,-DM
.	D 15,-DM
...Labyrinth	
...Spukhaus	
...Assembler	
TI 99/4	K 5,-DM
...Asteroiden	

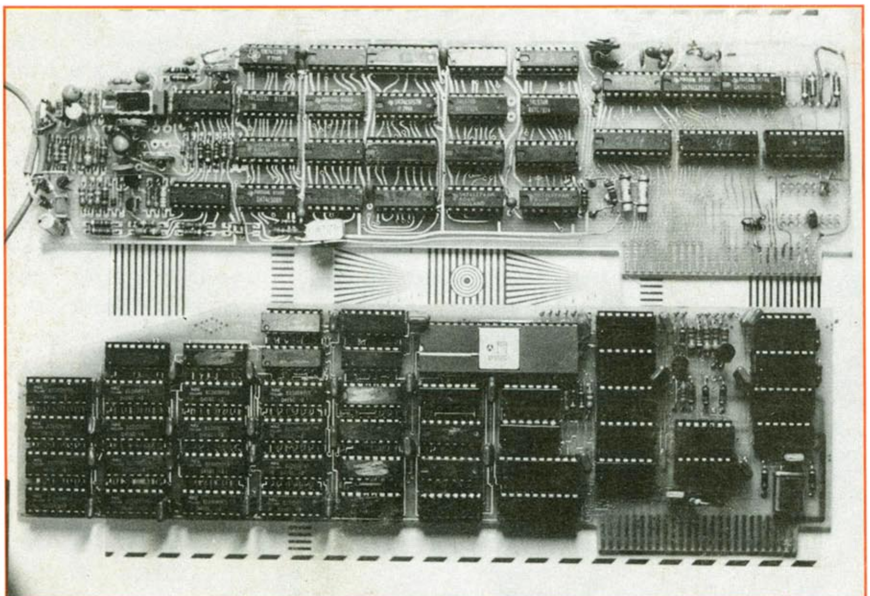




BARTSCHER ELEKTRONIK

Rüsten Sie Ihren Apple auf mit unseren Video-Interfaces!

(auch für Apple kompatible Computer)



AD 16-3
Echtzeitdigitalisierer

1356,– DM
incl. MWSt.

HGR-512
Graphicinterface

1356,– DM
incl. MWSt.

HGR-512 Graphicinterface

- Eigener 32 k Byte Bildspeicher
- Graphic-Controller EF 9365
- Umschaltbar 512 x 512 Punkte in schwarz-weiß oder 256 x 256 in 16 Graustufen bzw. 16 Farben auf jedem dieser Punkte
- Hardcopymöglichkeit von 512er-Modus
- ASCII-Zeichensatz in Größe und Richtung programmierbar
- Vektorplot mit 1 Millionen Punkte pro Sekunde
- Ausgang für RGB- und BAS-Monitore
- Demosoftware auf Diskette

AD 16-3 Echtzeitdigitalisierer

- Echtzeitdigitalisierer in 40 m sec.
- ca. 65 000 Bildpunkte mit 16 Graustufen
- eingebauter DA-Wandler zur Betrachtung des digitalisierten Bildes
- Digitalisierung von Videorecorder, Kamera oder Fernseher
- Keine Software für Bildlesevorgang nötig

NEUANKÜNDIGUNG:

AD 16-4

Echtzeitdigitizer mit
Sofortdisplay und
eigenem Bildspeicher

Digitalisierung und Display geschieht in 1/50 sec.
Karte enthält eigenen Bildspeicher.
Auflösung: 256 x 256 Bildpunkte.
Fremdsynchronisation der Kamera notwendig.



mit AD16-3 digitalisiert durch HGR-512 dargestellt

Bartscher Elektronik · Westring 84 · 3440 Eschwege · Tel.: 0 56 51 / 17 95